2014-1542, -1543

United States Court of Appeals for the Federal Circuit

MICROSOFT CORPORATION,

Appellant,

v.

PROXYCONN, INC.,

Cross-Appellant,

v.

MICHELLE K. LEE,
Deputy Director, U.S. Patent and Trademark Office,

Intervenor.

Appeals from the United States Patent and Trademark Office, Patent Trial and Appeal Board in Nos. IPR2012-00026 and IPR2013-00109.

BRIEF FOR CROSS-APPELLANT PROXYCONN, INC.

Matthew L. Cutler
Bryan K. Wheelock
HARNESS DICKEY & PIERCE, P.L.C.
7700 Bonhomme, Suite 400
St. Louis, MO 63105
(314) 726-7500
mcutler@hdp.com
bwheelock@hdp.com

Counsel for Cross-Appellant

SEPTEMBER 22, 2014

Form 9

FORM 9. Certificate of Interest

UNITED STATES COURT OF APPEALS FOR THE FEDERAL CIRCUIT

Microsoft Corporation v. Proxyconn, Inc.

No. 14-1542

CERTIFICATE OF INTEREST

Counsel for the (petitioner) (appellant) (respondent) (appellee) (amicus) (name of party) Proxyconn, Inc. (Cross Appellant) certifies the following (use "None" if applicable; use extra sheets if necessary):

1. The full name of every party or amicus represented by me is:
Proxyconn, Inc.

2. The name of the real party in interest (if the party named in the caption is not the real party in interest) represented by me is:
None

3. All parent corporations and any publicly held companies that own 10 percent or more of the stock of the party or amicus curiae represented by me are:
None

4. ☐ The names of all law firms and the partners or associates that appeared for the party or amicus now represented by me in the trial court or agency or are expected to appear in this court are:
Matthew L. Cutler of Harness Dickey & Pierece, P.L.C.; Bryan K. Wheelock of Harness Dickey & Pierce P.L.C.

June 26, 2014
Date

/s/ Matthew L. Cutler
Signature of counsel

Matthew L. Cutler
Printed name of counsel

Please Note: All questions must be answered
cc: John D. Vandenberg; Stephen Joncus

TABLE OF CONTENTS

Page

TABLE OF AUTHORITIES

Page(s)

Cases

Statutes

Other Authorities

STATEMENT OF RELATED CASES

Appellee/Cross-Appellant, Proxyconn, Inc., states, pursuant to F. Cir. R. 47.5, that (a) no appeal in or from the same proceeding was previously before this or any other appellate court; and (b) no case is known that will directly affect or will be directly affected by this Court's decision in the pending appeal.

JURISDICTIONAL STATEMENT

This action arises from an Inter Partes Review ("IPR") proceeding, Nos. IPR2012-00026 and IPR2013-000109, before the Patent Trial and Appeals Board of the U.S. Patent and Trademark Office ("Board"). The Board had jurisdiction over the IPR, pursuant to 35 U.S.C. § 314.

This appeal is taken from a Final Written Decision issued by the Board, on February 19, 2014, pursuant to 35 U.S.C. § 318 and 37 C.F.R. § 42.73. This Court has jurisdiction over this appeal pursuant to 35 U.S.C. § 319, 35 U.S.C. § 141(c), and 28 U.S.C. § 1295(a)(4)(A).

STATEMENT OF THE ISSUES

This appeal presents several issues related to the Board's incorrect determination of unpatentability of claims of U.S. Patent No. 6,757,717 ("The '717 Patent), including:

1. Did the Board legally err by adopting unreasonably broad constructions of the "sender/computer" and "receiver/computer," which encompass intermediate computers, i.e., computers expressly disclosed in the '717 Patent as

being separate and different than the sender/computer and the receiver/computer, and in view of that legal error, is the Board's decision that Claims 1, 3, 10, 22 and 23 of the '717 Patent are unpatentable, under § 102, over Santos erroneous? *See* sections B.1, B. 2, *infra*.

2. Did the Board legally err in adopting an unreasonably broad construction of "a gateway . . . between at least two other computers," which encompasses *any* two computers, including the separately recited caching computer, and in view of that legal error, is the Board's decision that Claims 6, 7 and 9 of the '717 Patent are unpatentable, under § 102, over DRP erroneous? *See* section B. 3, *infra*.

3. Did the Board legally err by adopting unreasonably broad constructions of the "sender/computer," which includes separate and independent computers, i.e., computers expressly disclosed in the '717 Patent as being separate and different than the sender/computer and the receiver/computer, and in view of that legal error, is the Board's decision that Claim 10 of the '717 Patent is unpatentable, under § 102, over Yohe erroneous? *See* sections C. 1, C. 2, *infra*.

4. Did the Board err in finding that Perlman is analogous art to the '717 Patent, and in view of that error, is the Board's combination of Perlman and Yohe to render Claim 10 of the '717 unpatentable legally erroneous? *See* section D. 1, *infra*.

5. Did the Board legally err by adopting an unreasonably broad construction of the "sender/computer," which includes elements of other separately described and claimed computers, and in view of that legal error, is the Board's decision that Claim 10 of the '717 Patent is unpatentable, under § 103, over Perlman and Yohe erroneous? *See* section D. 2, *infra*.

6. Did the Board legally err in shifting the burden to Proxyconn in order to find that Santos discloses permanent storage memory, and in view of that error, is the Board's decision that Claims 22 and 23 of the '717 Patent are unpatentable, under § 102, over Santos clearly erroneous? *See* section E. 1, *infra*.

7. Did the Board commit clear error in finding that merely looking at data, as disclosed in Santos, constitutes searching in predetermined locations in permanent storage memory, and in view of that error, is the Board's decision that Claim 23 of the '717 Patent is unpatentable, under § 102, over Santos clearly erroneous? *See* section E. 2, *infra*.

8. Did the Board commit clear error in finding that the GET function to a server, as disclosed in DRP, constitutes a *response* signal even though the server is unaware it relates to any other interaction, and in view of that error, is the Board's decision that Claims 11, 12 and 14 of the '717 Patent are unpatentable, under § 102, over DRP clearly erroneous? *See* section F., *infra*.

9. Did the Board err in denying Proxyconn's motion to amend by applying a heightened burden, retroactively promulgated after the filing of Proxyconn's motion to amend and inconsistent with 37 C.F.R. § 42.121, and further legally err in its determination that substitute claim in the '717 Patent? *See* section G., *infra*.

10. Did the Board correctly construe "searching" in claims 22-24 to mean "searching," and not "comparing" which is used to describe and claim different embodiments, so that its decision that Microsoft's Petition failed to demonstrate the unpatentability of claim 24 is correct and should be affirmed? *See* section H., *infra*.

STATEMENT OF THE CASE

This appeal arises from an IPR instituted by the Board under the America Invents Act ("AIA") for U.S. Patent No. 6,757,717. Appellee/Cross-Appellant, Proxyconn, is the assignee of the '717 Patent.

A. Procedural History

On September 18, 2012, Microsoft filed a Petition for *inter partes* review of Claims 1, 3, 10-12, 14 and 22-24 of the '717 Patent. A00089-134. This Petition asserted six different grounds of unpatentability against the '717 Patent. *Id*. Proxyconn waived its Patent Owner's Preliminary Response to the Petition. On December 21, 2012, the Board instituted an IPR as to Petition Claims 1, 3, 10 and

22-24, but not as to Claims 11, 12, and 14. A00329-55. In particular, the Board instituted review on only the following grounds: anticipation of Claims 1, 3, and 22-24 by U.S. Patent No. 5,742,820 ("Perlman"), anticipation of Claims 1, 3, 10, 22 and 23 by U.S. Patent No. 5,835,943 ("Yohe"), anticipation of Claims 1, 3, 22, and 23 by Santos ("Increasing Effective Link Bandwidth by Suppressing Replicated Date," Proceedings of USENIX Annual Conference, June 1998), and obviousness of Claims 1, 3, 10, and 22-24 in view of Perlman and Yohe. *Id*.

On January 11, 2013, Microsoft filed a second petition for *inter partes* review of Claims 6, 7, 9, 11, 12 and 14 of the '717 Patent. A00356-88. Microsoft asserted five additional grounds of unpatentability against the '717 Patent. *Id*. On January 11, 2013, Microsoft further filed a Motion for Joinder, pursuant to 37 C.F.R. § 42.122. On February 25, 2013, the Board instituted an IPR as to Claims 6, 7, 9, 11, 12 and 14. A00402-23. In particular, the Board instituted review on the following grounds: anticipation of Claims 6, 7, 9, 11, 12 and 14 by DRP (HTTP DISTRIBUTION AND REPLICATION PROTOCOL, W3C Note (August 25, 1997) http://www.www3.org/TR/NOTE-drp-19970825), anticipation of Claims 6 and 7 by Yohe, and obviousness of Claims 6, 7, 9, 11, 12 and 14 in view of DRP and U.S. Patent No. 6,292,880 ("Mattis"). *Id*. The Board also agreed to join the IPRs. A00424-29.

Proxyconn filed its Patent Owner's Response and Motion to Amend on May 21, 2013. A00430-48 and A00449-97. A corrected version of each was filed in June 20, 2013. A00562-80 and A00581-630. Microsoft replied to the Patent Owner's Response and opposed Proxyconn's Motion to Amend on August 21, 2013. A00631-49. Proxyconn filed its Reply to Microsoft's Opposition to its Motion to Amend on September 23, 2013. A00948-55. On November 18, 2013, an oral hearing was held. A01045-145.

The Board issued its Final Written Decision ("decision") on February 19, 2014. A00001-65. In its decision, the Board found Claims 1, 3, 6, 7, 9-12, 14, 22 and 23 to be unpatentable as anticipated and Claims 1, 3 and 10 to be unpatentable as obvious. The Board further determined that Microsoft failed to establish unpatentability as to Claim 24. Microsoft and Proxyconn each now appeal the Board's decision.

B. Statement of Facts

1. The '717 Patent

The '717 Patent relates to systems and methods for increased data access in a packet switched network. A00070 at Abstract. The invention provides increased data access by sending "digital digests" for data, rather than the actual data. A00080 at 2:17-25. When the data had been previously sent, the digital digest can

be used to identify the data, so that redundant transmission of the data can be avoided. This idea is used in a variety of ways in the '717 Patent. *Id*.

In one example embodiment illustrated in FIG. 4 reproduced below, a receiver sends a request for data to a sender. In turn, the sender locates the data, calculates a digital digest on the data, and sends the digital digest to the receiver. The receiver then searches in its "network cache memory" for data with the same digital digest. If it finds the data, it uses the data as if received from the sender and sends a positive indication signal back to the sender. In this manner, the sender, i.e., the source at one end of the data transmission, avoids sending the data to the receiver, i.e., the user at the other end. If, however, the receiver does not find the data, it sends a negative indication signal, which prompts the sender to send the data. A00073 at FIGS. 4-5, A00083 at 7:51-67.

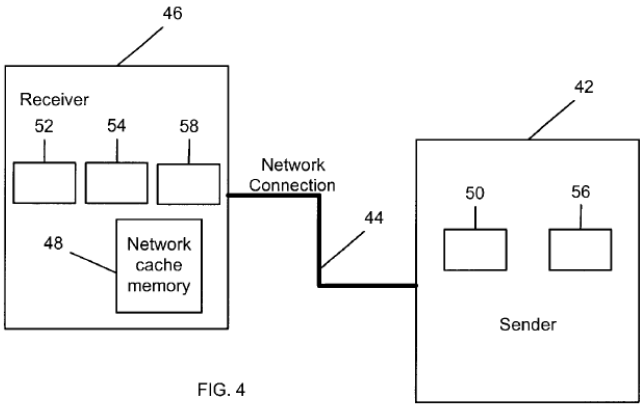


FIG. 4

A00073, FIG. 4

In an alternate embodiment, a gateway and caching computer are interposed between the sender and the receiver, for example, as shown in FIG. 11, reproduced

below. Because of their location, the gateway can intercept the negative indication signal sent from the receiver. Then, the caching computer searches for data with the digital digest from the negative indication signal, and if found (because it previously stored the data when sent from the sender), the data is sent to the receiver, and the negative indication signal is changed to a positive indication signal and sent along to the sender. A00077-78 at FIGS. 11-12, A00084 at 9:10-9:25. In this manner, the gateway and caching computers cooperate, as intermediaries, to avoid repeated transmission of the data from the sender, i.e., source.

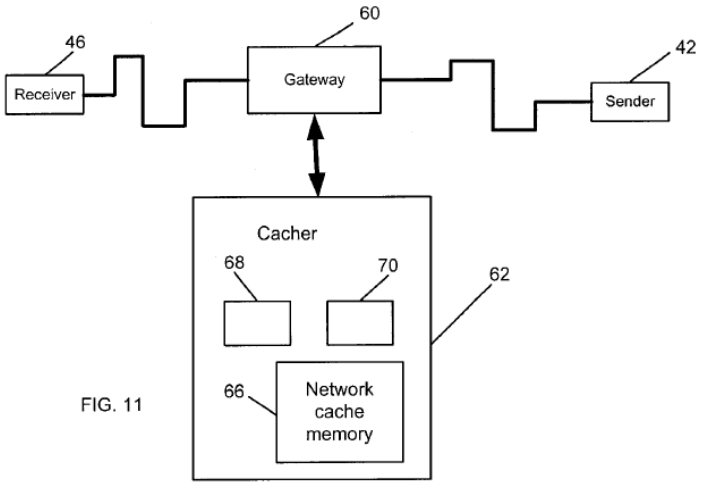


FIG. 11

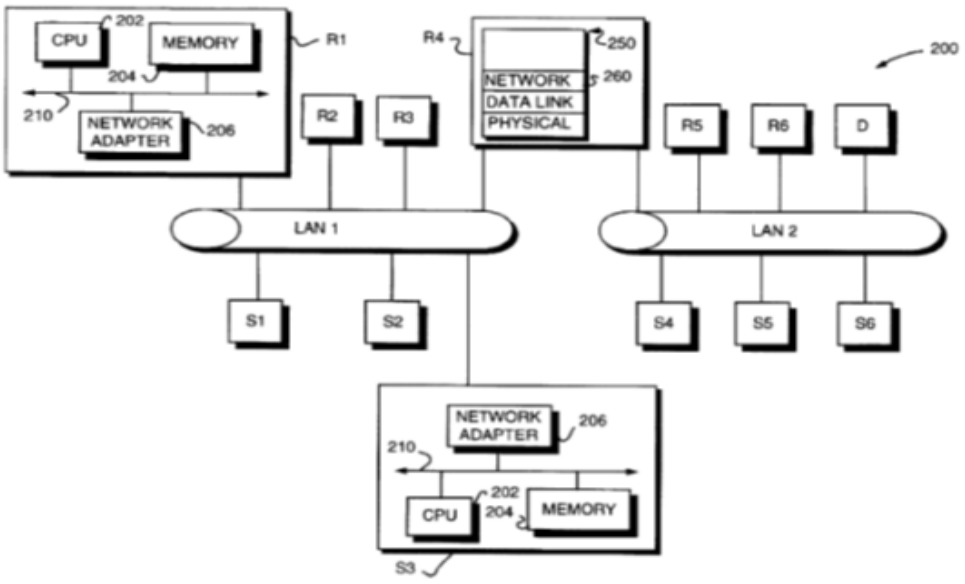
A00077, FIG. 11

The claims of the ‘717 Patent recite particular systems and methods for data access in a packet switched network. Claim 1 is exemplary of the claims of the ‘717 Patent, and recites:

> 1. A system for data access in a packet-switched network, comprising:
>
> a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through said network;
>
> said sender/computer further including means for creating digital digests on data;
>
> said receiver/computer further including a network cache memory and means for creating digital digests on data in said network cache memory; and
>
> said receiver/computer including means for comparison between digital digests.

2. Perlman

Perlman discloses a computer network for efficient synchronization of information across the network. A00142 at 1:6-8. As shown in Fig. 2 of Perlman, reproduced below, the network includes multiple routers (R1-6), including a designated router and other routers. Each router constructs a link state packet (LSP) containing information needed to generate a complete map of the computer network. A00137.



A00137, FIG. 2

The designated router generates a database identifier for its LSP and periodically broadcasts the identifier to all the other routers–not to any one router in particular. A00145 at 7:24-30. The identifier is broadcasted, based on a time interval, and not based on any request for data from any router. As such, Perlman discloses consistent and constant use of network bandwidth for broadcasting database identifiers to all routers based on a schedule rather than a request from other routers.

Upon receipt of the identifier, each router computes an identifier based on the router's content and compares the computed identifier to the received identifier. The router does not search for one identifier among several identifiers in memory. A00145 at 7:55-63. If the identifiers conform to one another, only the received identifier is stored. A00145 at 7:66-8:1. Otherwise, the router may request

the CSNP, or computer sequence packet, to resolve any differences. Each node or router of Perlman includes a central processing unit (CPU) and a memory unit 204, which consists of random access memory (RAM), i.e., a volatile, non-permanent memory. A00144 at 5:43-48.

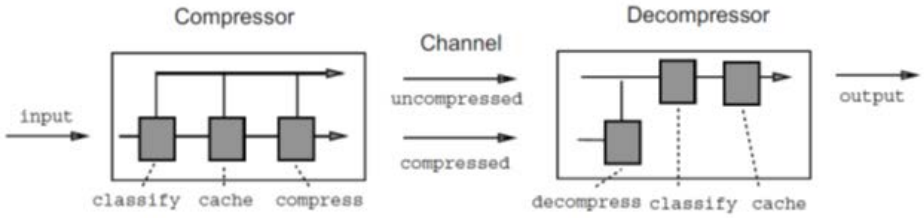
3. Yohe

Yohe describes a file-oriented distributed network. The network includes a remote client computer 12, which communicates with a file server computer 18 through a communication server 16. A00162. A cache verifying computer 14 is also disclosed as a separate, distinct computer, which is connected to the file server computer 18 through a local area network (LAN) 20. *Id.* While Yohe discloses that the cache verifying computer 14 may be combined with the communication server 16 (A00175 at 5:34-39), there is no similar statement that the cache verifying computer 14 should or even could be combined with the file server computer 18.

While permanent memories 34 and 80 are associated with the remote client 12 and the file server computer 18, respectively, none of the operations related to generating or comparing signatures are described with reference to memories 34 and 80, or data stored in such memories. Memory 34 of the remote client 12, for example, is only referred to as storing "hit ratios" for successful verification of data in cache memory.

4. Santos

Santos describes a link compression scheme that uses a memory-based cache to detect and remove redundancy at the packet level. A00148 at 14:18-23. The implementation of the scheme includes a compressor and a decompressor inserted in a network, i.e., gateways or other intermediate devices. The compressor intercepts input traffic, identifies a fingerprint for the data payload of the traffic, if possible, and transmits the fingerprint. In turn, the decompressor receives the fingerprint, removes compression, and transmits it as output traffic, as it was received by the compressor. A00153-54 at § 3.2.1.



A00152, FIG. 3

The compressor generates fingerprints on data being transferred through the network, not on data stored in any volatile or non-volatile memory. A00153-54 at § 3.2.1. Likewise, the decompressor generates a fingerprint for data transferred from the compressor, not on data stored in any volatile or non-volatile memory. Only after the fingerprint is generated, the decompressor stores the data indexed by the fingerprint in its volatile, RAM memory. *Id.*, A00155 at § 3.4.

Santos does not disclose permanent storage memory. In fact, Santos discloses that the "fingerprints" and "data payloads" are stored in RAM, such that the fingerprints and associated data payloads are lost during reset, e.g., as a power cycle or restart. *See* A00155 at § 3.3.

5. DRP

DRP discloses a protocol that uses a content identifier as uniquely identifying a piece of data or content. A00391-92. The content identifiers are compiled into an index, which is representative of a hierarchical structure of files, such as an HTML page. A00392-93. The index is stored at the HTTP server ("sender"). A00393-94. If a client ("receiver") wants to view the HTML page, the client retrieves or downloads the index, and then automatically downloads the files that are specified in the index. A00393-95.

Once the initial download is completed, the client can update the HTML page by downloading a new version of the index, and comparing it against the previous version of the index. A00394-95. Simply stated, the client uses the index as verification that it has the valid files to construct the HTML page. If the verification fails, the client downloads only the file or files that are different, via a GET request. *Id.* The client does not transmit any signal containing positive, partial or negative indication. The server, in turn, does not respond to any signal from the client.

SUMMARY OF ARGUMENT

The Board's findings of unpatentability contain numerous legal errors that stem from the Board's attempt to use claim construction to remedy the shortcomings of the Petitions filed by Microsoft.

The Board's reliance on BRI, and not *Phillips*, is improper in IPR proceedings, because there is no meaningful opportunity to amend the claims when the BRI differs from the true meaning. An IPR proceeding is not examination, in which amendments may be freely made, but is an adjudicative review of the issued U.S. patent, to which a presumption of validity attaches. In fact, the Board directs patent owners to pursue amendments to issued claims through alternative proceedings, i.e., reexamination or reissue applications. *Innolux Corp. v. Semiconductor Energy Lab. Co., Ltd*., IPR2013-00066, Paper 24 (July 18, 2013) at p. 5. *See* Section B.1, *infra*.

To the extent BRI is the standard, the Board failed to construe the claims consistent with the broadest reasonable interpretations ***in view of the specification***; rather, the Board adopted ***unreasonably*** broad constructions, which are at odds with the specification. The constructions adopted by the Board improperly reshape the claims of the '717 Patent to encompass the prior art. Specifically:

- construing "sender/computer" and "receiver/computer" in Claims 1, 3, 10, 22-23 to include separately disclosed intermediaries or separate,

independent computers, is incorrect as a matter of law (*see* Sections B.2 and C, *infra*); and

- construing "at least two other computers" in Claim 6 to include the caching computers, separately recited in Claim 6, is incorrect as a matter of law (*see* Section B.3, *infra*).

Because the prior art does not teach these limitations *when properly construed*, the Board erred in finding: Claim 10 anticipated by Yohe, Claims 1, 3, 10, 22 and 23 anticipated by Santos, and Claims 6, 7, and 9 anticipated by DRP.

Further, the Board relies on Microsoft's framing of the problem of reducing redundancy to recast the *synchronization* problem actually dealt with in Perlman. There is no reason for a person of ordinary skill to look to synchronization as disclosed in Perlman when faced with a redundant transfer problem, without the guidance of the '717 Patent. Perlman is not an analogous art, and it is therefore clearly erroneous for the Board to combine Perlman and Yohe as it did. *See* Section D.2, *infra.* Moreover, in its analysis, the Board erred by finding that one "could" combine Perlman and Yohe, rather than any reason why one skilled in the art "would" combine the references, which is the proper inquiry. The Board's determination that Claim 10 is unpatentable over Yohe and Perlman, based on this legal error, is improper. *See* Section D.2, *infra.*

In addition, the Board commits legal error by shifting the burden to Proxyconn to show that a permanent storage memory, as required by Claim 22, is not disclosed in Santos. Microsoft simply asserts that a Pentium II machine "necessarily" includes ROM because it is not expressly disclosed. Microsoft does not submit expert testimony on this point, yet the Board found its arguments sufficient to shift the burden to Proxyconn. In response Proxyconn explains, with citation to its expert, that permanent storage memory is not disclosed in Santos. The Board determined Proxyconn's explanation was "inconclusive" and then determined in favor of Microsoft. *The lack of evidence demonstrates that the burden, contrary to 35 U.S.C § 316, was on Proxyconn, not Microsoft*. The same conclusory assertion is presented by Microsoft as to Claim 23. Again, the Board challenges Proxyconn to show "finding" a fingerprint is not a search in a predetermined location. There is no disclosure of any particular location in permanent storage memory in Santos, much less permanent storage memory. Again, the Board assumes the parts of Santos supports Microsoft's Petition, which they do not, and looks for Proxyconn to establish patentability. This is legal error. Further, in the absence of a legal error, in its decision regarding Claims 22 and 23, the Board clearly erred in finding that Santos discloses: i) the permanent storage memory and ii) searching in *predetermined* locations in permanent storage memory, as explained below. *See* Section E, *infra.*

In addition to the above legal errors, the Board clearly erred in its understanding of the prior art cited in the petitions. In particular, DRP is the description of a protocol, which is implemented in servers and clients. The protocol permits a client to realize which files it is missing and request those files from the server. The request for files, or GET function, as explained below, is simply a file request to the server — and not a "response" signal. The server is oblivious to content the client is ultimately assembling, why it wants a particular file, and importantly, whether the file request relates to any prior interaction with the client. The Board's determination that a mere file request is a "response" signal is clearly erroneous. *See* Section F, *infra.*

Finally, the Board erred in applying an improper standard to deny Proxyconn's motion to amend. In particular, the Board cites to the *Idle Free* decision, a decision not yet rendered at the time Proxyconn's motion to amend was filed, and lists conditions the Board considers necessary for a patent owner to "meet" its burden. A00054-55. These conditions, however, are plainly erroneous and inconsistent with the clear conditions prescribed by Patent Office regulation, 37 C.F.R. § 42.121, which outlines the conditions under which the Board *may* deny a motion to amend. By departing from the regulation, and its erroneous finding regarding the scope of substitute Claim 36, the Board committed legal error. *See* Section G, *infra*

The Board correctly determined that the paper construction of searching is searching. The Board was correct in rejecting Microsoft's arguments that searching should be limited to comparing a separate term used in separate embodiments, to describe a different activity. *See* Section H, *infra.*

ARGUMENT

A. Standard for Review

The Board's legal conclusions and statutory interpretation are reviewed *de novo*. *Belkin Int'l, Inc. v. Kappos*, 696 F.3d 1379, 1381 (Fed. Cir. 2012). Anticipation and prior art teachings present questions of fact. *In re NTP, Inc*., 654 F.3d 1279, 1297 (Fed. Cir. 2011). The determination of obviousness under 35 U.S.C. § 103(a) is a legal conclusion based on underlying findings of fact. *In re Mettke*, 570 F.3d 1356, 1358 (Fed. Cir. 2009). The Board's ultimate determination of obviousness is reviewed *de novo*. *In re Kotzab*, 217 F.3d 1365, 1369 (Fed. Cir. 2000). This Court reviews the Board's factual findings for substantial evidence. *In re Gleave*, 560 F.3d 1331, 1335 (Fed. Cir. 2009).

This Court does not defer to the Board's interpretation of Patent Office regulation, when it is plainly erroneous or inconsistent with the regulation. *In re Garner*, 508 F.3d 1376, 1378-79 (Fed. Cir. 2007).

B. The Board Erred in Construing the Claims of the '717 Patent

1. *It is Legal Error to Use the Broadest Reasonable Interpretation or "BRI" in Inter Partes Review Proceedings*

35 U.S.C. § 2(b) permits the Patent Office to establish regulations, not inconsistent with law, which shall govern the conduct of proceedings in the Patent Office. Separately, the America Invents Act ("AIA"), passed in 2011, empowers the Patent Office to prescribe regulations governing conduct of IPRs. 35 U.S.C. § 316. Nothing in the AIA, however, changes the limitation on the Patent Office's authority under § 2, i.e., regulations established by the Patent Office 1) must *not* be inconsistent with law, and 2) must be limited to the "conduct" of the proceedings. Under the AIA, the Patent Office promulgated regulations to govern IPRs. These regulations include 37 C.F.R. § 42.100(b), which mandates that a challenged patent be given "its broadest reasonable construction in light of the specification of the patent in which it appears." This regulation exceeds the Patent Office's authority under § 2.

First, regulation § 42.100(b) is inconsistent with law. Issued patents are property rights, which enjoy a presumption of validity under 35 U.S.C. § 282. The presumption of validity is commensurate with the actual scope of the patent. That scope is not a "nose of wax, which may be turned and twisted in any direction"—by a patentee, a third party, or even the Patent Office. *White v. Dunbar*, 119 U.S. 47, 51 (1886). To adjudicate the validity of an issued patent, the Patent Office ***must***

consider the actual scope of the patent. By considering a different scope, such as the fictitious scope created by BRI, the adjudication ignores the presumption of validity, and reduces the issued patent to a patent application, much like a reissue application (i.e., an application for further examination *voluntarily filed by a Patent Owner*). This is clearly inconsistent with law, i.e., 35 U.S.C. § 282.[1]

Second, "conduct of the proceeding" under 35 U.S.C. § 2 has been interpreted by this Court to permit "procedural" rule making, but not "substantive" rule making. *Cooper Techs. Co. v. Dudas*, 536 F.3d 1330, 1335 (Fed. Cir. 2008); *Tafas v. Doll*, 559 F.3d 1345, 1352 (Fed. Cir. 2009) *vacated*, 328 F. App'x 658 (Fed. Cir. 2009) (per curiam). A rule is substantive if it changes the standards by which the Patent Office examines an application. *Tafas*, 559 F.3d at 1353-54. By construing claims according to BRI, and not according to *Phillips*, the Patent Office alters the scope and meaning of issued patents, and ultimately, alters validity of the patents. In effect, 37 C.F.R. § 42.100(b) sets the stage for the Patent Office to declare a patent invalid under BRI, when a district court would not invalidate that same patent by under its proper *Phillips* construction. This is an

[1] The Board has expressly stated the presumption of validity under 35 U.S.C. § 282 are inapplicable to the Patent Office, citing only to the difference in the standards of proof. *SAP America v. Versata Dev. Grp., Inc.*, CBM2012-00001, Paper 70 (June 11, 2013 at p. 7). This is incorrect. Regardless of the standard of proof, ***the burden remains on Petitioner to show a challenged patent is invalid***. No statute or regulation disturbs the presumption under 35 U.S.C. § 282, even for patents challenged in an IPR proceeding.

improper termination of an issued patent–a government-granted property right of the Patent Owner. There can be no realistic argument that 37 C.F.R. § 42.100(b) is something other than a substantive rule, i.e., a rule exceeding the Patent Office's authority under § 2.

The Patent Office has been permitted by this Court to apply BRI as a claim clarifying tool during examination, in lieu of the principles outlined in *Phillips*. The BRI approach serves a public interest in "reducing the possibility that claims, finally allowed, will be given broader scope than is justified." *In re Yamamoto*, 740 F.2d 1569, 1571 (Fed. Cir. 1984). Patent applicants' interests are only unaffected, under BRI, when the applicants are "not foreclosed from obtaining appropriate coverage for their invention with express claim language." *Id.* at 1571 (Fed. Cir. 1984); *In re Skvorecz*, 580 F.3d 1262, 1267 (Fed. Cir. 2009) ("[BRI's] purpose is to facilitate exploring the metes and bounds to which the applicant may be entitled, and thus to aid in sharpening and clarifying the claims during the application stage, when claims are readily changed.").

Stated simply, because applicants may freely amend claims during examination proceedings, the BRI approach is appropriate. If the scope of the claims under BRI, as understood by the Patent Office during examination, is broader than intended, the applicant has the ability – and the right – to freely amend the words of the claims ***so that the proper interpretation is expressed***. The

unfettered ability to amend claims reduces the BRI, in examination proceedings, to a procedural rule.

Conversely, IPR is not an examination proceeding. *Abbott Labs v. Cordis Corp.*, 710 F.3d 1318, 1326 (Fed. Circ. 2013); H.R., 112-98—Part 1, page 46-7 ("The Act converts inter partes reexamination from an examinational to an adjudicative proceeding, and renames the proceeding 'inter partes review'"). IPR is nothing like examination because the Patent Owner has no right, and in practice little opportunity, to amend or otherwise clarify the scope of the terms of the issued claims. The Patent Office provides for a "motion to amend" under 37 C.F.R. §316. Specifically, the Patent Owner may file "1 motion to amend the patent" to cancel challenged claims or propose a reasonable number of "substitute" claims for each challenged claim. 37 C.F.R. §316. The Patent Office is free to deny the motion to amend, when the Patent Owner does not "establish that it is entitled to the requested relief." 37 C.F.R. § 42.20(c).

In *Idle Free,* the Board detailed the burden of the Patent Owner to show entitlement to amend its claims. *Idle Free Systems v. Bergstrom, Inc.*, IPR2012-00027, Paper 26, (June 11, 2013). Specifically, the Board indicated that a Patent Owner is required to present a complete listing of substitute claims (showing markups over the issued claims), provide citations to the original disclosure and earlier-filed disclosures for each claim that is added or amended, offer claim

constructions for new claim terms, show patentability over the prior art of record, show patentability over ***all*** prior art known to the Patent Owner, and further make some representation "about the specific technical disclosure of the closest prior art." *Id*. at p. 7. The Board further demands supporting expert testimony, because mere statements by counsel are "inadequate." *Id*. at p.7-8.

Along with this heighted burden, the Patent Owner is afforded a mere 15 pages, in 14 pt. font, double-spaced, to demonstrate its entitlement to the substitute claims. While a motion for additional pages can be made, as of the filing of this paper, Proxyconn is not aware that the Board has granted any motion seeking additional pages. In view of the above, it is clear that amendments to the claims are a practical impossibility during IPR. Informatively, as of the filing of this paper, only one (1) motion to amend has been granted (*i.e*., a motion submitted on behalf of the U.S. government). *Int'l Flavors & Fragrances Inc. v. U.S. Dept. of Ag.*, IPR2013-00124, Paper 12 (May 20, 2014). Indeed, the Board has suggested that if a Patent Owner wants to amend its claims, it should consider a reissue or reexamination. *Innolux Corp. v. Semiconductor Energy Lab. Co., Ltd*., IPR2013-00066, Paper 24 (July 18, 2013 at p. 5).

Under the conditions created by the Patent Office and the Board, the motion to amend provided by Congress, and the only arguable support for the BRI

approach, is a nullity.[2] No reasonable opportunity exists to amend claims during IPR. As such, there is no mechanism, much less a "readily" available mechanism, for a Patent Owner to sharpen or clarify the meaning of claim terms when they are distorted by BRI. *Skvorecz*, 580 F.3d at 1267. Without that opportunity, the Board adjudicates the validity of the issued patent based on a fictitious, broader scope, thereby ignoring the presumption of validity afforded to issued patents under 35 U.S.C. § 282.

The Board's decision, because it applies the BRI approach to the issued claims of the '717 Patent, should be reversed.

2. The Board's Construction of "Sender/Computer" and "Receiver/ Computer" is Incorrect under BRI and Phillips, and Caused its Incorrect Determination of Unpatentability of Claims 1, 3, 10, 22, and 23 based on Santos

In its decision, the Board construed the sender/computer ("sender") to mean a computer that sends data and a receiver/computer ("receiver") to mean a computer that receives data, each computer can encompass multiple devices *including intermediaries*. A00015. The '717 Patent, however, includes an express

[2] Even if a motion to amend was possible, which it is not, the Board is equipped to separately apply the BRI approach to "substitute" claims presented in a motion to amend, while still applying the principles of *Phillips* to issued claims. There is no issue with the Board applying the *Phillips* approach, because it is already required to do so, based on the status of the claims. See 37 C.F.R. §42.100(b) (only unexpired claims are construed under BRI) and MPEP § 2258 G. As indicated above, the *Idle Free* decision from the Board mandates that the Patent Owner offer claim constructions in its motion to amend. *Idle Free Systems v. Bergstrom, Inc*., IPR2012-00027, Paper 26 (June 11, 2013) at p. 7.

disclosure of intermediaries as being separate and different from the sender and the receiver, i.e., gateways and caching computers. The Board ignores this aspect of the '717 Patent and reshapes "sender" and "receiver" to capture the ***intermediaries*** disclosed in the prior art. This is incorrect. In light of the specification, the sender is reasonably construed to mean a source of data (within the network) that sends the data, and the receiver is reasonably construed to mean a computer that receives and uses the data. ***Each can include multiple computers, even separate housings, but not computers separately identified and performing distinct roles within the system.***

Starting with the claims of the '717 Patent, Claim 1, for example, is directed to a system for data access and recites:

> **1**. A system for data access in a packet-switched network, comprising:
>
> a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through said network;
>
> said sender/computer further including means for creating digital digests on data;
>
> said receiver/computer further including a network cache memory and means for creating digital digests on data in said network cache memory; and
>
> said receiver/computer including means for comparison between digital digests.

As shown, the system of Claim 1 is recited in terms of a "sender/computer" and a "receiver/computer." Conversely, Claim 6 is also directed to a system for data access and recites:

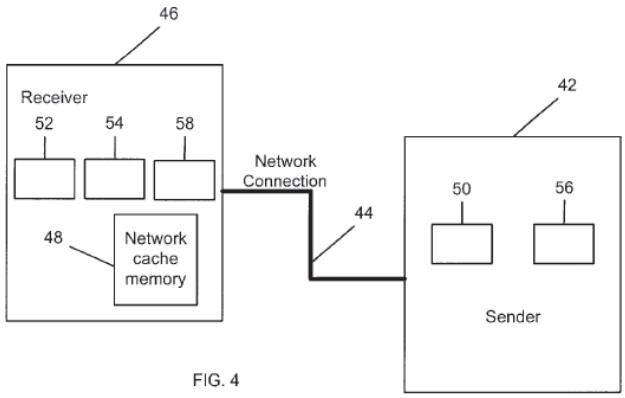
> **6**. A system for data access in a packet-switched network, comprising:
>
> a gateway including an operating unit, a memory and a processor connected to said packet-switched network in such a way that network packets sent between at least two other computers pass through it;
>
> a caching computer connected to said gateway through a fast local network, wherein said caching computer includes an operating unit, a first memory, a permanent storage memory and a processor;
>
> said caching computer further including a network cache memory in its permanent storage memory, means for a digital digest and means for comparison between a digital digest on data in its network cache memory and a digital digest received from said packet-switched network through said gateway.

As shown, the system of Claim 6 is recited in terms of a "gateway" connected between at least two "other" computers and a "caching computer." The remaining claims use these four "computer" terms as the building blocks of the systems and methods in the '717 Patent. By referring to these "computers" by different terms, the '717 Patent invokes a presumption that each of the claim terms means something different. *Nystrom v. TREX Co.*, 424 F.3d 1136, 1143 (Fed. Cir. 2005).

The remainder of the specification of the '717 is dispositive of the meaning of these terms. The Background introduces the sender and receiver, stating the "client (receiver) 4 caches data received from network 2" and a "remote server (sender) 8" sends data over the network. A00080 at 1:19-25, *see* A00071 at FIG. 1. The use of the terms permeates the specification. In several embodiments, the

sender and receiver exchange digital digests for data, rather than the actual data, in an attempt to avoid transmitting the data from the sender to the receiver. In particular, if the receiver already has data with a digital digest that matches the digital digest of the requested data, there is no need to transmit the data from the sender to the receiver and the receiver is able to use the data as if received from the sender. A00080 at 2:18-52.

FIG. 4 of the '717 Patent, reproduced below, illustrates one embodiment, in which a system includes a sender 42 and a receiver 46.



A00073, FIG. 4

The interaction between the sender 42 and receiver 46 is depicted in a different figure, i.e., A00073 at FIG. 5. The sender 42 calculates a digital digest on the data in means 50 and then transmits the calculated digest to receiver 46. A00083 at 7:52-55. The receiver 46 receives the digital digest and then searches its memory for data with the same digest. A00083 at 7:55-57. If it finds the data with the same digest, ***it uses the data as if it were received from the sender 42*** and

sends a positive indication signal to the sender 42. A00083 at 7:57-60. If not, the receiver 46 sends a negative indication signal to the sender 42. A00083 at 7:60-61. Upon receiving a negative indication signal, the sender 42 transmits the data. A00083 at 7:61-63. This transaction begins with a receiver 46 sending a request to the sender 42. A00083 at 7:65-67.

In this embodiment, and others in the '717 Patent, there is no suggestion that the receiver sends the data to other computers – rather it "uses" the data. A00080 at, e.g., 2:22-25 ("the receiver/computer can ***use*** the data immediately. . .")(emphasis added), 2:28-31 ("If the receiver/computer already has data with the same digital digest, it ***uses*** this data. . . .")(emphasis added), 7:57-59 ("If [the receiver] finds such data, it uses it as if it were received from the sender/computer 42. . .").

Simply put, the sender 42 is one end of the data transaction, and the receiver 46 is the other end of the data transaction. The sender and receiver, in the context of the '717 Patent, are ***NOT*** intermediate computers. However, the '717 Patent expressly provides for certain intermediate computers, i.e., gateways and caching computers. *See*, e.g., A00077; A00083-84 at 8:57-9:30.

As shown in FIG. 11, reproduced below, a gateway 60 can be interposed between the sender 42 and receiver 46. A00077 at FIG. 11; *see also* A00071 at FIG. 2. The '717 Patent defines the term "gateway" as including "network proxies

and routers." A00080 at 2:14-15. The gateway 60 can be connected to a caching computer 62. This alternate embodiment, however, is missing from the Board's description of the '717 Patent. *See* A00003-07.

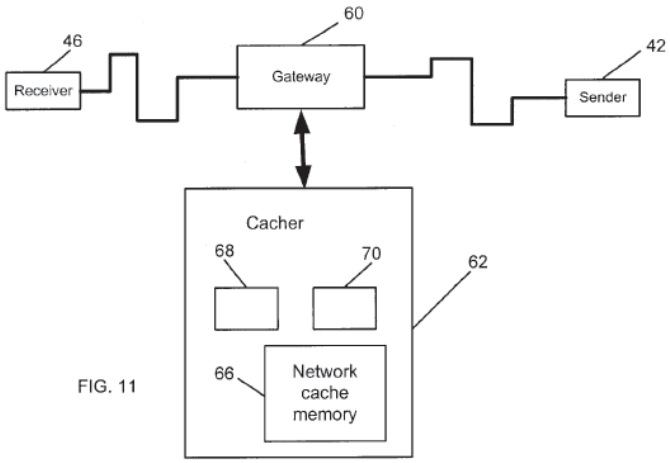


FIG. 11

A00077, FIG. 11

The operation of the system of FIG. 11 is described with reference to FIG. 12. Initially, the gateway 60 intercepts data sent from the sender 42 to the receiver 46, saves it in its memory, and passes it unchanged to the receiver 46. A00084 at 9:13-16. Then, if the gateway 60 intercepts an indication signal other than a positive indication signal issued by the receiver 46 (i.e., indicating the receiver 46 does not have the data, A00084 at 9:17-19), the caching computer 62 searches for data with the same digital digest in its memory. A00084 at 9:19-21. If that digest is found, then the gateway 60 sends the data to the receiver 46, changes the indication signal to positive, and then passes the indication signal to sender 42. A00084 at 9:21-24. Accordingly, as described, the gateway 60 sends data, and separately

receives data. Yet, nowhere in the '717 Patent is the gateway referred to as a sender or a receiver --- ***because it's not an END of the data transmission; it's not the source or the user of the data to be transmitted***. It would unreasonably expand the terms "sender" and "receiver" to re-label the gateway 60 as either a sender or a receiver.

It is clear that the sender 42 is separate from the receiver 46. Likewise, the sender 42 is separate from a gateway 60, and the receiver 46 is separate from the gateway 60. And further, the caching computer 62 is separate from the receiver 46 and the sender 42. The same is not necessarily true of the gateway 60 and the caching computer 62, because the caching computer 62 is the cache of the gateway 60 – both being intermediate between the sender 42 and the receiver 46. The '717 Patent further ***expressly*** states that "the gateway computer 60 may be integrally formed with the caching computer." A00084 at 9:6-7. The '717 Patent makes no similar statement regarding integration of the caching computer 62 or the gateway 60 with other parts in FIG. 11, *i.e.*, the sender 42 and the receiver 46.

The Board reads the above-cited excerpt to mean that the sender 42 and receiver 46 *may* include intermediaries. A00017. This is unreasonable. The '717 Patent illustrates the caching computer 62 and the gateway 60 as discrete parts between the sender and the receiver. *See* A00077 at FIG. 11. Because they are illustrated as separate blocks, the '717 Patent must expressly disclose it is possible

to integrate the two computers for that to be reasonably understood. The logical extension of this statement is that other discrete parts are not integrated, i.e., the receiver, for example, *cannot* be integral with the caching computer.

Apart from the above, the Board further cites to a "collection of functionally defined subsystems 48, 52, and 54" of the sender 42, and injects "housing" as a possible delineation. Specifically, the Board indicates the '717 Patent conveys to the skilled artisan that the described computers may or may not be in "separate housings." A00017.. Though the '717 Patent does not use the term "housing," Proxyconn has not taken the position that the sender or the receiver cannot include separate housings. Rather, the distinction exits in the ***separate function*** of the different computers. As an example, if a network includes a web server (or sender), a file server (or sender), a router for each floor of an office building, and also 15 clients (or receivers) per floor, no skilled artisan would lump the router into one of the clients, or one of the servers. These are separate parts of the network that ***function independent of one another***. It is unreasonable to permit the definition of sender or receiver to include intermediate computers, such as, for example, gateways, proxies, routers, and caching computers, because ***they are separate and independently functional parts of a network***.

In finding to the contrary, the Board's decision diverges from the plain disclosure of the '717 Patent to expand the terms "sender/computer" and

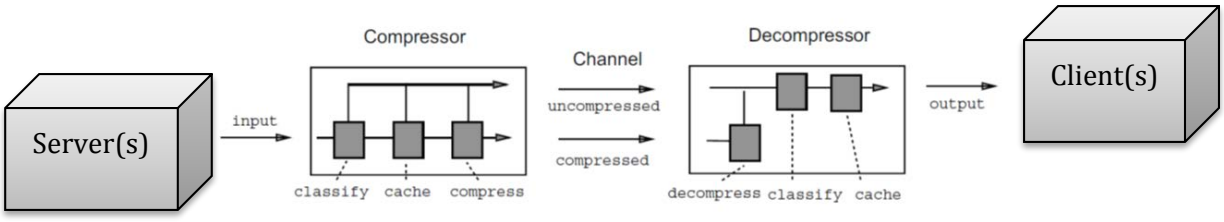
"receiver/computer" to encompass the "intermediaries," including the separately disclosed gateway or caching computers. The Board's error in claim construction enlarged the scope of Claims 1, 3, 10[3], 22 and 23 of the '717 Patent, leading to the Board's erroneous determination that these claims are anticipated by Santos.

More specifically, Santos discloses a "compressor" and a "decompressor." In the context of the '717 Patent, these devices are most closely akin to the gateway and caching computers. *Compare* A00152 at § 3.1. ("The compressor is a repeater (***perhaps part of a router***) that accepts input traffic.") and A00080 at 2:14-15 ("The term 'gateway' as used herein also includes ***network proxies and routers***.") (emphasis added).

---

[3] Despite its assertion to the contrary, the Board did not institute a ground of unpatentability based anticipation of Claim 10 by Santos. For each ground of review in the Petition, which became a part of this IPR, the Board expressly found, as required by 37 C.FR. § 42.108(c), that a "reasonable likelihood of prevailing" existed. The Board made no such determination as to Santos and Claim 10. At pages 19-20 of Paper 17, the Board made its determination regarding Claim 10 and other references. Yet, nowhere, in this section or elsewhere in Paper 17, does the Board decide that a reasonable likelihood of claim 10 being unpatentable over Santos existed. Because the § 42.108(c) clearly states that the Board "shall" not institute unless there is such a finding, this ground of unpatentability is not part of the IPR proceeding. Despite its omission, the Board faults Proxyconn for not addressing Santos, specifically as to Claim 10. In support of its position, the Board makes no citation to its Decision to Institute (Paper 17), because there is, in fact, no finding regarding Claim 10 and Santos sufficient under §42.108(c). Because the Board failed to institute on this ground of unpatentability, its determination of unpatentability of Claim 10 based on Santos is improper. For this reason alone, the Board's determination as to Claim 10 and Santos should be reversed. Apart from this reason, Claim 10 recites a sender/computer and a receiver/computer. As such, the Board's decision rests on the erroneous claims construction of these terms, and should be reversed for that reason.

This parallel is further demonstrated by the annotated figure below. The Board did not simply reconstruct the sender, for example, to capture a separate computer acting as part of the sender. Rather, as applied to Santos, the Board all but ignores the "server" and "client" in Santos, and focuses on the separate intermediate "compressor" and "decompressor" – which are routers separate from the server or client and function independently, actually acting as a go-between for "many" servers and clients. A00149-50 at § 2.1, A00152 at FIG. 3 and § 3.1.



A00152, FIG. 3, annotated with Server(s) and Client(s)

With reference to the figure above, the decompressor, for example, intercepts packets with "fingerprints" and replaces the fingerprints with data, before forwarding the packet to the network, i.e., onto to the client. A00153 at § 3.2.1. The compressor and decompressor are intended to be transparent to the servers and the clients. A00149-50 at § 2.1(implemented in a "diverse environment hosting many servers and clients,"); *see also* A00148 at § 1 ("It utilizes a source of correlation that is not available at individual clients and servers.. . ." and ". . . caching in a transparent manner. . ."). The compressor and decompressor operate

separate and independently from the servers and clients of the network in which they are implemented. Neither is an end point in the transmission of packets in the network; neither is the source of packets to be sent; and neither is the *user* of the packets sent. As such, neither is a receiver or sender in the context of the '717 Patent. Santos, therefore, fails to disclose each and every limitation of Claims 1, 3, 10, 22 and 23 of the '717 Patent.

Proxyconn contends that the proper constructions, under *Phillips*, of "sender/computer" and "receiver/computer" *exclude* intermediaries. Yet to the extent this Court finds that BRI is indeed the appropriate standard, the construction must still be reasonable and thus ***still exclude intermediaries***. Thus, under either standard of construction, the Board's constructions are legally erroneous, and its determination that Claims 1, 3, 10, 22 and 23 are unpatentable under § 102 based on Santos should be reversed.

3. The Board's Construction of "Gateway . . . Between at Least Two Other Computers" is Incorrect under BRI and Phillips, and Caused its Incorrect Determination of Unpatentability of Claims 6, 7 and 9 based on DRP

There are at least two other computers recited in Claim 6 in addition to the gateway computer and the caching computer. Despite the plain language of the claim, the Board construed the "at least two other computers" in Claim 6 to be "any two other computers" and further determined the "any two other computers" can include the separately recited caching computer. A00017-18 and A00047. The

Board's novel construction, rendered for the first time in its decision, is inconsistent with the claims and the detailed description of the '717 Patent as explained below. The construction appears to be designed by the Board to fill gaps in the Petition and support its determination that Claim 6 is anticipated by DRP. Specifically, the Board uses the façade of BRI to reduce the number of computers required by Claim 6, not because that is what is *reasonable* in view of the '717 Patent, but because that is all that is disclosed in DRP.

> **6**. A system for data access in a packet-switched network, comprising:
>
> a gateway including an operating unit, a memory and a processor connected to said packet-switched network in such a way that network packets sent between at least two other computers pass through it;
>
> a caching computer connected to said gateway through a fast local network, wherein said caching computer includes an operating unit, a first memory, a permanent storage memory and a processor;

Claim 6 clearly and unambiguously recites the "two other computers" as separate from the gateway and the caching computer. The claim goes so far as to include the term "other" to avoid any doubt that the "other" computers are different than the recited gateway and caching computer.

Further, Claim 6 recites the gateway is "connected" to the network, which includes at least two other computers so that network packets pass through it. Additionally, Claim 6 recites that *the gateway is connected to the caching computer*. Thus two connections are recited. If one of the two computers was the caching computer, as determined in the Board's decision, the recited "connection"

between the gateway and the caching computer would be redundant of the connection between the gateway and the "other" computer. Claims should not be construed to make terms superfluous. *Merck & Co. v. Teva Pharms. USA, Inc*., 395 F.3d 1364, 1372 (Fed. Cir. 2005) ("A claim construction that gives meaning to all the terms of the claim is preferred over one that does not do so."); *Phillips v. AWH Corp.*, 415 F.3d 1303, 1314 (Fed. Cir. 2005)("the context in which a term is used in the asserted claim can be highly instructive.").

Furthermore, Claim 8 recites that the "caching computer is integrally formed with said gateway." A00083. The Board suggests that the term "other" is limited to the manner in which the gateway is connected. A00018. Even if that were true, Claim 8 further recites the caching computer is integrally formed with the gateway. This proves the caching computer cannot be one of the "other" computers because it is part of the gateway in Claim 8, and not an "other" computer connected to it. *In re Suitco Surface, Inc*., 603 F.3d 1255, 1260 (Fed. Cir. 2010) ("The broadest-construction rubric coupled with the term . . . does not give the PTO an unfettered license to interpret claims to embrace anything remotely related to the claimed invention. Rather, claims should always be read in light of the specification and teachings in the underlying patent.")

The '717 Patent further informs the meaning of "two other computers." For example, FIG. 11, reproduced below, includes a system in which a gateway 60 is interposed "between at least two other computers 42 and 46." A00083 at 8:64-66.

Notably, the references 42 and 46 correspond to the sender/computer and the receiver/computer, respectively. *See* A00071 at FIG. 11 and A00083 at 7:20-27. The caching computer 62 is connected to the gateway 60.

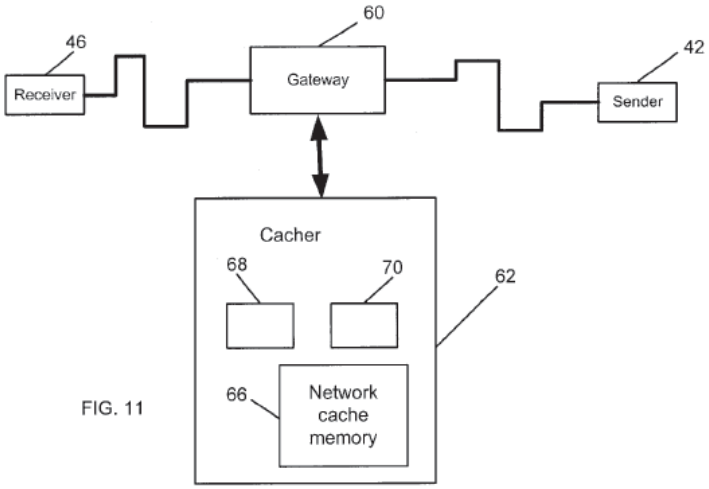


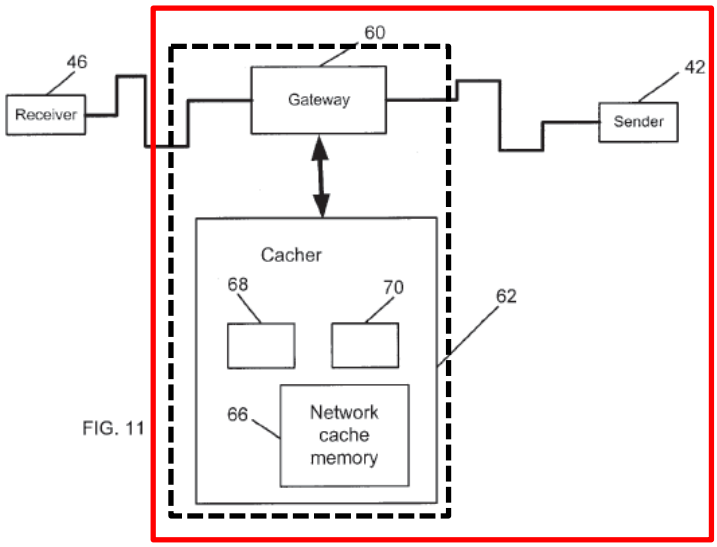
FIG. 11

A00077, FIG.11

The point of the gateway and caching computers is for the gateway to send data requested by the receiver 46, thereby alleviating the need for the sender 42 to send the data. The intermediate position of the gateway 60 and caching computer 62 between the two other computers (e.g., the sender 42 and receiver 46) permits it to perform this fundamental function. If, as the Board intends, the caching computer is the sender, the description of the gateway 60 and caching computer 62, and their function, becomes pointless because there is no intermediate computer possessing the data. The example collapses to a sender and receiver, as illustrated in FIG. 4.

Finally, as indicated with Claim 8 above, the '717 Patent expressly states that the "gateway 60 may be integrally formed with the caching computer."

A00084 at 9:6-7. FIG. 11 has been modified with a dotted box, below, to illustrate the gateway 60 and caching computer 62 integrally formed. A00077



A00077, FIG. 11 (annotated)

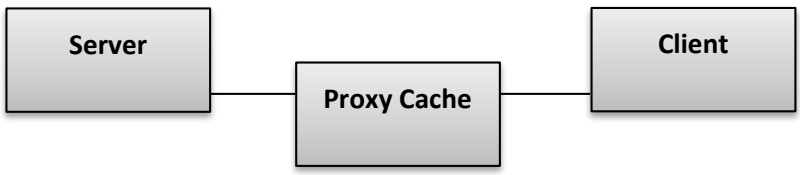
Now, if the Board's construction is applied, the caching computer 62 is one of the "other" computers, as indicated by the RED block. By the disclosure of the '717 Patent, ***the Board's construction collapses the system illustrated in FIG. 11 to just <u>two</u> computers: the receiver 46 and a combination computer (***i.e.***, the sender 42, the gateway 60, and the caching computer 62).*** This is not a reasonable construction, and fails to give meaning to the term "other" in Claim 6. *Merck & Co.*, 395 F.3d at 1372 ("A claim construction that gives meaning to all the terms of the claim is preferred over one that does not do so.").

In view of the above, the Board's construction, which permits the caching computer recited in Claim 6 to be one of the "other" computers, is unreasonably

broad. The erroneous construction forms the basis for the Board's conclusion that Claims 6, 7 and 9 are anticipated by DRP.

DRP is a description of a distributed and replication protocol to improve the efficiency and reliability of data distribution over HTTP. The protocol is described with reference to three parts: a client, a server (i.e., an HTTP server) and a proxy cache. A00393, e.g., 5:22-32 and A00397 at 9:38-10:2. No structure is disclosed in DRP for the server, the proxy cache, or the client, with one exception. DRP notes that "a client can use a disk cache for data files." A00395 at 7:3.

The protocol is implemented to provide efficient transfer of data from the server to the client, especially when the client requests the same data multiple times. A00390 at 2:23-30. When present, the proxy cache, by definition, is an intermediary between the server and the client, whereby data sent between the server and the client passes through the proxy cache. A00398 at 10:37-39.



[This drawing is created for illustration Only; it is <u>NOT</u> included in DRP]

From the above synopsis, and more generally DRP, it is clear that the only potential "gateway" as recited in Claim 6 of the '717 Patent is the proxy cache, which is connected between two "other" computers, i.e., the client and the server,

whereby data sent between the client and the server passes through the proxy cache. Nothing else in DRP is disclosed consistent with a gateway between two other computers. On this basis, the proxy cache is the gateway, and the server and client are the two "other" computers.

To determine whether the proxy cache may further be considered the caching computers (as expressly permitted in the '717 Patent), the recited limitations of Claim 6 are considered. First, DRP does not disclose that the proxy cache includes permanent storage memory. In arguing to the contrary, Microsoft only cites to DRP's disclosure that a "proxy may cache indexes if the HTTP header indicates that this is allowed." A00394 at 6:32-33; A00400 at Appendix A. This excerpt is silent as to where the indexes may be "cached" and whether it is in persistent or non-persistent memory. Microsoft submits no expert testimony that one skilled in the art would understand the mere reference to "cache" to indicate permanent storage memory. As such, Microsoft fails to show that the proxy cache includes permanent storage memory. And, the Board makes no finding that the disclosed proxy cache includes permanent storage memory.

Second, DRP does not disclose the proxy cache as including a "means for calculating." In its Petition, Microsoft fails to cite to *any* part of the DRP as disclosing the proxy cache as having a "means for calculating." Microsoft instead cites only to the "client" and the "server" as including the recited means -- each is

separate from the proxy cache. As such, there is no argument that the proxy cache includes "means for calculating," and, the Board makes no such finding. For these two reasons, *the proxy cache cannot be the caching compute r as recited in Claim 6, and thus DRP fails to anticipate Claim 6*.

This should be the end of the Board's analysis. Except, the Board reshapes Claim 6, so that the "other" computer is also the separately recited caching computer. This is unreasonable. For the reasons outlined above, the '717 Patent does not contemplate the caching computer is one of the two "other" computers. Rather, the '717 Patent makes clear that the gateway and caching computers are intermediaries, which function to reduce the communication between the two "other" computers. They may even be integrated together, ***but they cannot be the "other" computers.*** The Board's construction is erroneous under BRI and certainly under *Phillips*.

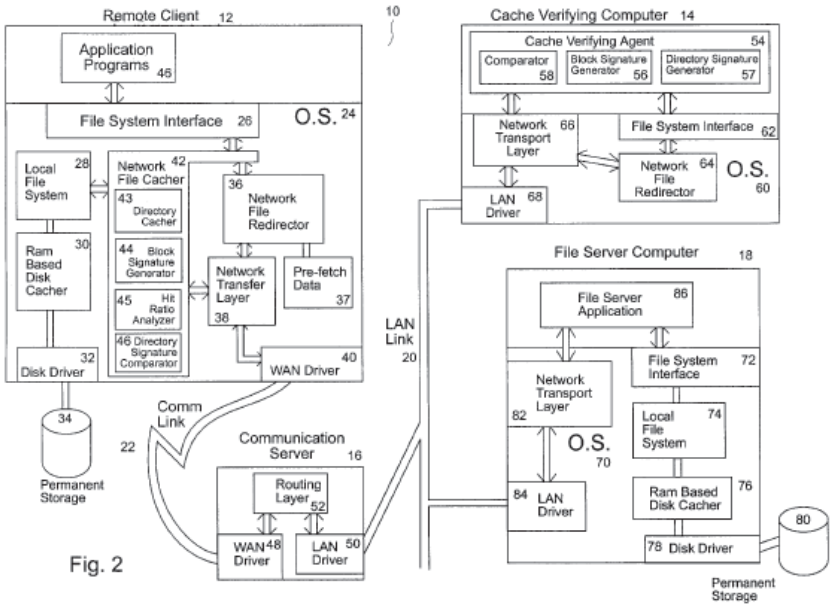
Proxyconn contends the proper constructions, under *Phillips*, of "at least two other computers" precludes the separately-recited caching computer from being one of the "other" computers. Yet, to the extent this Court finds that BRI is indeed the appropriate standard, the construction must still be reasonable and still exclude the caching computer of Claim 6. As such, the Board's determination that Claims 6, and Claims 7 and 9 depending therefrom, are unpatentable under §102 based on DRP should be reversed.

C. The Board Erred in Finding Claim 10 is Anticipated by Yohe

> **10**. A system for data access in a packet-switched network, comprising:
>
> a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through a network;
>
> said sender/computer further including means for creating digital digests on data, and
>
> said receiver/computer further including a network cache memory, means for storing a digital digest received from said network in its permanent storage memory and means for comparison between digital digests.

1. Yohe Does Not Have a Sender/Computer as used in Properly Construed Claim 10

The Board concluded that sender/computer may broadly encompass "multiple devices." A00036. In particular, the Board appears to have adopted Microsoft's argument that the Yohe's cache verifying computer 14 and file server 18 constitute the required sender/computer.



A00024, FIG. 2

The Board's interpretation is incorrect because it does more than not limit sender/ computer to hardware residing in one housing, but it raids other separately described and identified devices to find missing components needed to anticipate the claims. The cache verifying computer and file server are shown, described, and claimed as separate devices in Yohe. Similarly, a caching computer is a separate element from the sender/computer in the lexicon of the '717 Patent. See, for example caching computer 62 in Fig. 11:

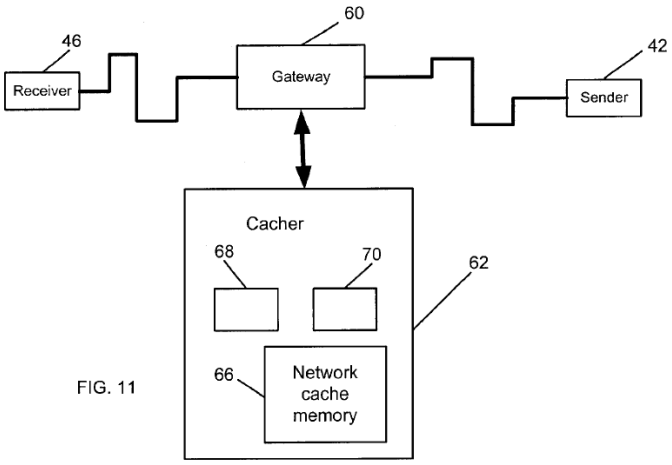


FIG. 11

A00077, FIG. 11

In order to anticipate a claimed invention, a prior art reference must not only disclose all elements of the claim within the four corners of the document but must also disclose those elements arranged or combined in the same way as in the claim. *Net MoneyIN, Inc. v. VeriSign*, 545 F.3d 1359, 1370-71 (Fed. Cir. 2008). By construing Claim 10 beyond its ordinary and plain meaning under *Phillips*, to an unreasonably broad construction, the Board has improperly expanded anticipation to the point where the claim elements simply have to be in the same document. The Board's erroneous claim construction has led to the erroneous conclusion that Yohe has a sender computer as claimed in Claim 10 of the '717 Patent.

2. Yohe Does Not Have a Receiver/Computer as used in Properly Construed Claim 10

The Board determined that "means for comparison between digital digests" means a "structure that can compare any digital digest received from the network

with another digital digest. A00038. However "digital digests" in the context of the '717 Patent, and Claim 10 in particular are digital digests of data:

> said sender/computer further including means for creating digital digests on data, and

While the remote client computer 12 of Yohe includes a directory signature comparator 46, it does not disclose a comparator for signatures created on data. Yohe therefore lacks a receiver/computer with means for comparison between digital digests on data. In the context of the '717 Patent, and Claim 10 in particular, the Board's construction of a comparator for comparing any digital digest, even a type of digital digest not disclosed in the '717 Patent is unreasonably broad. Yohe does not disclose "means for comparison between digital digests" when this term is given its ordinary and plain meaning according to *Phillips*, or at least its broadest reasonable interpretation. A00038.

D. The Board Erred in Finding Claims 1, 3, and 10 Unpatentable as Obvious from Yohe and Perlman

1. The Board's Finding that Perlman was Analogous Prior Art is Clearly Erroneous

In erroneously concluding that Perlman is analogous prior art to the '717 Patent, the Board was persuaded by testimony from Microsoft's expert that Perlman, Yohe, and the '717 Patent all address the same problem, namely: "the desire to reduce redundancy in network data transmissions where dynamic data previous sent over the network has been stored by the receiver for possible later

use." A00050. However, this statement of the problem is colored by Proxyconn's claimed solution.

As Proxyconn explained to the Board, the problem addressed by the '717 Patent is "increasing the speed of data accessing." A00080 at 1:10-15. Conversely, the problem addressed by Perlman was the "efficient *synchronization* of information across a computer network." A00142 at 1:8-9. When properly approached from the perspective of a person of ordinary skill in the art at the time of the invention, Perlman's efficient synchronization of information simply would not have suggested itself as relevant to speeding data access. As explained by Proxyconn's expert, a person of ordinary skill in the art would dismiss Perlman's synchronizing information across a network as unnecessarily burdening network (slowing, rather than speeding data accessing), and unnecessarily requiring information replication at each node. A00050.

It is only after awareness of Proxyconn's claimed solution– the use of digital digests to speed data access– that Perlman, which uses database identifiers, seems relevant. The Board's reliance on Microsoft's hindsight framing of the problem resulted in clearly erroneous determination that Perlman's data synchronization was relevant prior art to the '717 Patent.

2. The Board's Finding that Claims 1, 3, and 10 Would have been Obvious over Perlman and Yohe is Incorrect

> **1**. A system for data access in a packet-switched network, comprising:
>
> a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through said network;
>
> said sender/computer further including means for creating digital digests on data;
>
> said receiver/computer further including a network cache memory and means for creating digital digests on data in said network cache memory; and
>
> said receiver/computer including means for comparison between digital digests.
>
> **3**. The system as claimed in claim **1**, wherein said receiver/computer further includes means for storing said created digital digest in its first or permanent memory.

The Board found that it would have been obvious to incorporate Yohe's permanent storage memory into Perlman's Router. A00051. The Board correctly stated that the test for obvious is not whether the features can be bodily incorporated into the primary reference, but what the combined teachings would have suggested to those of ordinary skill in the art. *Id.* However, the Board identified no reason why a person of ordinary skill would incorporate Yohe's memory in Perlman's router. The Board cites testimony of Microsoft's expert (A00192-95), but this testimony simply suggests that a person of ordinary skill could combine the teachings, and gave no real reason why they would. In fact, Microsoft's expert impermissibly relies on the disclosure of the '717 Patent.

A00195. Microsoft's expert improperly put the burden on the '717 Patent to demonstrate unexpected results (A00195), and the Board echoes this when it said that "Proxyconn proffers no persuasive evidence in support of its position (A00051), rather than demanding proof of some reason why a person of ordinary skill (not an expert like Dr. Long) would have made the combination.

The Board concedes that Yohe does not disclose means for comparison of digital digest on data located in the receiver/computer. A00051. The Board said that Perlman describes performing the comparison of digital digests on data in the receiver and concluded that it would have been obvious to incorporate Perlman's comparing means into Yohe's remote client. The Board's conclusion is erroneous because Perlman is not analogous prior art, and therefore, no matter what Perlman discloses, it cannot make the claimed invention obvious. *See In re Horn,* No. 79-546, 1979 WL 25011, at *2 (CCPA Nov. 15, 1979)("For no matter what a reference teaches, it could not have rendered obvious anything, 'at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains,' unless said hypothetical person would have considered it.").

The Board determined that incorporating Perlman's comparing means into Yohe's remote client, or moving Yohe's comparator 58 into the remote client would have involved nothing more than ordinary skill and would have been using

known devices to perform known functions to yield predictable results. A00052. However Board has confounded what a person of ordinary skill in the art *could* do with what would have been *obvious* to a person of ordinary skill in the art. The Board gives no reason why a person of ordinary skill in the art would provide the comparator on the remote client, rather than simply leaving it on the cache verifying computer where it can compare signatures from multiple remote clients with signatures of data in the cache. Without a reason to provide comparators on each of the remote clients, rather than on the single cache verifying computer, merely because it could be moved does not mean that it would have been obvious to do so. The Board erred in concluding that Claims 1, 3, and 10 would have been obvious.

E. The Board Erred in Finding "Santos" Anticipates Claims of the '717 Patent

1. Santos Does Not Have A Receiver/Computer with Permanent Storage Memory as Recited in Claim 22

Claim 22 is directed to a method for increased data access performed by a receiver/computer and recites:

> **22**. A method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of:
>
> receiving a message containing a digital digest from said network;
>
> searching for data with the same digital digest in said network cache memory,
>
> if data having the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting it back through said network; and
>
> creating a digital digest for data received from said network cache memory.

As explained in Section B.2 , *supra*, the Board's determination of anticipation by Santos is in error because the claim construction applied by the Board is unreasonable.

Additionally, the Board finds disclosure of a receiver/computer having permanent storage memory, as recited in Claim 22. *See* A00044-46, related to Claim 23. Rather, Santos discloses that its compressor and decompressor were implemented in "a pair of Intel-based Pentium II 300 MHz machines running Linux 2.0.31 with 128MB of RAM each." A00155 at § 3.4. To be clear, RAM is volatile memory, i.e., ***not*** permanent storage memory. Separately, Santos discloses (A00155):

3.3 Reset Messages

During normal operation of the protocol, the compressor keeps track of all illegal fingerprints (i.e., those fingerprints for which a collision occurred.) In the event that this state is lost (e.g., the compressor is restarted), the compressor reliably sends a cache reset message to the decompressor to ensure that the decompressor does not have any entries indexed by a previously illegal fingerprint.

Further, restarting the decompressor during operation of the protocol may result in significant rejection traffic. Therefore, we explicitly send a cache reset message from the decompressor to the compressor. This is merely a performance optimization, and is not essential for correctness.

This disclosure indicates that upon re-start, the decompressor experiences loss of fingerprints, thereby requiring a reset message to be sent to the compressor to clear its cache. The cache is then rebuilt as fingerprints pass between the compressor and decompressor. ***This shows that the fingerprints are being stored in volatile, not permanent, memory***. When a PC reboots, its RAM is cleared, but its hard drive (i.e., permanent memory) still includes program, document, video, and music files, for example. *This suggests that the cache in Santos is, therefore, not permanent storage memory.* In its decision related to Claim 23, the Board cites to the declaration of Dr. Konchitsky, Proxyconn's' expert, stating "[t]his indicates to me, as it would to any person of ordinary skill in the art[,] that fingerprints are

not stored in permanent storage memory." A00046. The Board erroneously determined Santos to be "inconclusive" regarding the character of the cache memory. A00045.

The Board then turns to Microsoft's argument. Specifically, Microsoft notes that, as stated above, the compressor and decompressor have 128MB of RAM, i.e., volatile memory. Microsoft further cites to Santos's disclosure that "we limited the amount of memory available for caches, excluding the overhead induced by the hash table implementations, to 200MB each." A00155 at § 3.4. Based on that, the Board indicates that because Santos expressly discloses only 128 MB of RAM, and the cache is limited to 200MB, the cache ***must*** be in non-volatile memory, i.e., permanent storage memory. This is, at best, an assumption to fill gaps in the disclosure of Santos. The '717 Patent is an issued patent, which includes Claim 23. The '717 Patent enjoys a presumption of validity, and a Petitioner in an IPR proceeding is required to show by a preponderance of the evidence that the patent is invalid. 35 U.S.C. §§ 282 and 316. The Board's ***assumption*** about the type of memory that is or is not included in a Pentium II machine cannot satisfy this standard, particularly when it contradicts the disclosure.

The Board's analysis leads to the conclusion that the Board, contrary to law, shifted the burden to Proxyconn to show patentability. This is improper. As mandated by 35 U.S.C. § 316, the "petitioner shall have the burden of proving a

proposition of unpatentability by a preponderance of the evidence." The burden is on and remains with Microsoft, and Microsoft failed to satisfy its burden of showing a permanent storage memory is disclosed in Santos. By shifting the burden to Proxyconn to show permanent storage memory is *not* disclosed in Santos, the Board legally erred and its decision regarding anticipation of Claim 22 by Santos should be reversed.

To the extent it did not legally err in this way, the Board clearly erred in finding that Santos discloses permanent storage memory, as required by Claim 22, and therefore the Board clearly erred in its determination that Santos anticipates Claim 22. The Board's determination regarding the unpatentability of Claim 22 and Claim 23 which depends therefrom, should be reversed.

2. Santos does Not Disclose Searching in Predetermined Locations as Required by Claim 23

Claim 23 is dependent from Claim 22 and recites:

> **23**. The method as claimed in claim **22**, further comprising searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to the digital digest received from said network.

The Board clearly erred in its determination that Claim 23 is invalid, because the Board improperly expanded Santos beyond the limits of its disclosure. In particular, the Board found that Santos discloses searching in ***predetermined*** locations. The Board indicates that Santos "looks at H(x)", i.e., a fingerprint and

also "looks at the stored payload associated with the H(x)." A00046. This is undeniably a disclosure of "looking" at data, but it is not a disclosure of searching in a *predetermined location*. There is no indication in Santos of "where" the cache is located, much less that the searching is directed to a "predetermined" location in permanent storage memory.

For the above reasons, the Board clearly erred in its determination that Santos discloses searching in predetermined locations, and the Board's determination that Santos anticipates Claim 23 is clearly erroneous. The Board's determination regarding the unpatentability of Claim 23 should be reversed.

F. The Board Erred in Finding the "DRP" Anticipates Claims of the '717 Patent

1. DRP Does Not Disclose "Receiving a Response Signal" As Recited in Claim 11

Claim 11 is directed to a method performed by a sender/computer and recites:

> **11**. A method performed by a sender/computer in a packet-switched network for increasing data access, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit data to a receiver/computer, the method comprising the steps of:
>
> creating and transmitting a digital digest of said data from said sender/computer to said receiver/computer;
>
> receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and
>
> if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer.

As explained in Section B.3 , *supra*, the Board's determination of anticipation by DRP is in error because the claim construction applied by the Board is unreasonable.

Further, the Board clearly erred in its determination that DRP anticipates Claim 11, because DRP fails to disclose receiving a response signal from the receiver/computer.

DRP discloses a client and a server. Rather than retrieving data in the first instance, the DRP protocol causes the client to retrieve an index for content to be received, such as an HTML page. The client then compares the index to a previous version of the index to determine which, if any, files have changed. A00393 at 5:30-33. In this way, the client determines the minimum set of files that it needs to download to bring the client up to date. *Id*. That is, the client may be in possession of some files needed to construct the HTML page, and therefore only need to download files it does not already have to construct the HTML page.

After identifying the minimum set of files, the client downloads those files from the server. The protocol is driven by the *client*. There is no disclosure or suggestion in DRP that the server, understands whether the files downloaded by the client are related to an index previously downloaded by the client, or not. In other words, the file download by the client is not linked to a prior index download, and therefore *CANNOT* be considered to be a "response." This is by

design in the DRP because it is designed to be backwards compatible with existing servers. A000398-99 at 10:43-11:8 ("The DRP protocol is defined so that it does not necessarily require any server support."). Basically, the client manages implementation of the DRP protocol because it cannot rely on the server as understanding the protocol. This is contrary to Claim 11, which recites a "method performed by the sender/computer." A00085 at 11:34.

In order for the server to be "receiving a response signal," the server in DRP has to be able to understand a response may be forthcoming, and correlate the original message or transmission to which the "response" is related. It does not. For example, one index may include 13 files, while another includes 75 files; each of which may be retrieved by different clients repeatedly. A separate download by a client for 3 particular files, it is not an indication that 10 files are not needed by one client, or that 72 files are not needed by the other client, because there is no disclosure or even a suggestion, in DRP, that the server understands or keeps track of what index the client is comparing. The server only understands that 3 files are needed -- the 3 files may or may not be all the files associated with an HTML page. It does not know. As such, DRP fails to disclose receiving a "response" signal from a receiver/computer. The request by the client is not a response to an inquiry, but an independent communication generated on the client's initiative.

The Board rejects the above argument citing to DRP's disclosure of a GET request, a differential GET request, and no request – each generated and sent by the client. The Board thus ignores the limitations of Claim 11, which is directed to the sender/computer and receiving a ***response*** signal. The server in DRP does not receive these requests as a "response" signal. As such, the Board's finding that DRP discloses a "response" signal is clearly erroneous.

For these reasons, the Board erred in its determination that DRP anticipates Claim 11, and the Board's determination that Claim 11, and Claims 12 and 14 dependent therefrom are unpatentable, should be reversed.

G. The Board Erred in Denying Proxyconn's Motion to Amend, as to Claims 35 and 36

The Patent Owner filed a motion to amend. The motion includes substitute Claims 35 and 36, which were based on Claims 1 and 3 of the '717 Patent. In presenting the substitute claims, the Patent Owner fulfilled all the regulatory requirements of a motion to amend, within the allotted pages, and its motion should have been granted.

The Board cites to several reasons for its denial of Proxyconn's motion, as it pertains to Claims 35 and 36. Specifically, the Board indicates that Proxyconn did not: construe the newly added claim terms, address the manner in which the claims are patentable generally over the art, identify the closest prior art known to it, address the level of ordinary skill in the art at the time of the invention, or discuss

how such a skilled artisan would have viewed the newly recited elements in light of what was known in the art. In support of its denial, the Board cites to its decision in *Idle Free*. Stating that it had issued before Proxyconn's motion to amend had been filed. This is incorrect. Proxyconn filed its motion to amend on May 21, 2013. A00430-48. The *Idle Free* decision was not rendered until June 11, 2013. [4]

In articulating these novel conditions to amend, the Board exceeded the Patent Office regulations, by which the Board is bound. Specifically, 37 C.F.R § 42.121 permits the Board to deny a motion to amend when it (i) does not respond to a ground of unpatentability involved in the trial, or (ii) the amendment seeks to enlarge the scope of the claims or introduce a new matter. This list is complete. It is not open-ended and does not invite the Board to ad-lib additional conditions, after the fact, under which it is permitted to deny motions to amend. The Board, however, has done just that. The Board's extra conditions are plainly erroneous, in view of § 42.121, and inconsistent with the express grounds of denial in that regulation. These conditions, therefore, cannot form a basis for the Board's denial of Proxyconn's Motion to Amend.

With regard to the conditions identified in 37 C.F.R § 42.121, the Proxyconn satisfied its burden. In fact, Claim 35 is intended to clarify the meaning of data

[4] Proxyconn did seek to amend its motion to amend for the limited purpose of correcting a few typos and citations. That was filed June 20, 2013, but was not intended nor permitted to make substantive changes to the motion to amend. A00562-80.

access, as argued by Proxyconn in the IPR. Specifically, Claim 35 is amended to additionally recite the receiver/computer as being configured to initiate a request for data. With the amendment, Proxyconn addresses each of the grounds of unpatentability involved in the trial. Claim 35 is a substitute for Claim 1 of the '717 Patent. The grounds of unpatentability in the IPR for Claim 1 consisted of: anticipation by Perlman, anticipation by Yohe, anticipation by Santos, and obviousness in view of Perlman and Yohe. Proxyconn addresses each of these grounds (i.e., presumably the closest prior art) in the Patent Owner's Motion to Amend, and within the 15 pages allotted. The Board fails to cite to any deficiency in the arguments presented. Rather, the Board relies on its list of "conditions" in rejecting substitute Claim 35, stating simply, Proxyconn has not carried its burden. The Board has not indicated how any of the conditions, specific to substitute Claim 35, affect its decision to deny the motion. For example, how would claim construction affect its decision? The Board, rather than deciding the motion on its merit, relies on an impossible list of conditions, which are inconsistent with § 42.121, to eliminate Proxyconn's opportunity to clarify the underlying claim. For these reasons, the Board erred in applying a heightened burden, which is plainly erroneous and inconsistent with 37 C.F.R § 42.121, to the motion to amend, and denying the motion to substitute Claims 35 and 36 based on that improper standard.

Further, with regard to substitute Claim 36, the Board denies the motion, because it deemed Claim 36 as broader than Claim 3 (i.e., the claim it is intended to replace). This is incorrect. The Board's conclusion is based on replacing "communicating through said network" with "configured to communicate with one another through said network." The later, by amendment, positively recites ***structure*** in the substitute Claim 36. The sender/computer and receiver/computer, as part of the system Claim 36, are now ***structured*** in this manner. Conversely, "communicating through said network" in the system claim 1 is a statement of intended use, and not a limitation of Claim 1. *Hewlett–Packard Co. v. Bausch & Lomb, Inc*., 909 F.2d 1464, 1468 (Fed. Cir. 1990) ("[A]pparatus claims cover what a device is, not what a device does."). Accordingly, the amendment creates limitation when none previously existed. This is clearly a narrowing amendment, not a broadening amendment. The Board legally erred in finding to the contrary.

For the above reasons, the Board legally erred its denial of Proxyconn's Motion to Amend, and therefore, the Board's decision should be reversed.

H. Microsoft's Petition does not Prove that Claim 24 is Unpatentable

Microsoft appeals the Board's determination that its Petition failed to prove that Claim 24 is unpatentable, contesting the Board's construction of "searching" to mean searching. Specifically, the Board construed the term "searching for data with the same digital digest in said network cache memory" according to its

ordinary meaning as requiring "an ability to identify a particular data object with the same digital digest from a set of potentially many data objects stored in the network cache memory."A00020-21. Instead, Microsoft argues that "searching" cannot possibly mean "searching," and instead searching for data means comparing two digests.

1. The Plain and Ordinary Meaning of Searching

The Board got it right. Searching means searching. After noting that "[t]he Specification never expressly defines 'search,'" the Board looked for the plain meaning of search, referring to several different dictionary definitions that it found "reflect that a skilled artisan would have understood 'search' to involve analyzing a set of items to identify one particular item from among a set of items." A00020-21. The Board's construction comports with the words of Claims 22-24: "searching for data with the same digital digest in said network cache memory." A00085 at 12:30-53. This construction also comports with the description of several embodiments, for example:

> The invention yet further provides a method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of receiving a message containing a digital digest from said network; searching for data with the same digital digest in said network cache memory, and if data having the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting it back through said network.

A00081 at 4:1-11. *See also*, *Id.* at 4:12-22; *Id.* at 4:23-38; A00082 at 5:1-11; A00083 at 7:56; *Id.* at 8:22, 27, 51; A00084 at 9:20; *Id.* at 10:6. Several of these uses show that searching in the context of the '717 Patent is more than a simple comparison between two digests. For example, in the embodiment described at A00083 at 8:50-53:

> on every digital digest. Still alternatively, receiver/computer **46** may search not only in its network cache memory **48**, but also in predefined locations in its permanent storage memory. Sender/computer **42** may add to a digest it sends to the receiver/computer information about the possible location of the data with that digital digest in the receiver/computer's permanent storage memory.

In this embodiment, the search for data is conducted in the network cache memory and locations in the permanent storage memory. This searching in multiple locations is completely consistent with the Board's construction of searching and searching and completely inconsistent with Microsoft's assertion that searching is a simple comparison.

2. The Board's Construction is Consistent with the Construed Meaning and the Broadest Reasonable Interpretation

Microsoft argues that the Board chooses a narrower interpretation "over the reasonable and broader" interpretation proffered by Microsoft. However as explained above, construing "searching" to mean "identifying a particular data object with the same digital digest from a set of potentially many data objects" is consistent with the ordinary meaning of searching, and with several embodiments

of the ‘717 Patent. In contrast, Microsoft’s proposed construction of searching: “comparing” is not the ordinary meaning of searching; and it is inconsistent with embodiments of the invention that describe searching.

There are embodiments that expressly describe comparing. A00083 at 7:35-36; A00084 at 9:27-28. The specification and claims of the ‘717 Patent use the words “searching” and “comparing” in two different contexts, and these two different words are presumed to have different meanings. *Nystrom*, 424 F.3d at 1143. The Board’s construction allows “searching” and “comparing” to have different meetings, Microsoft’s proposed construction would give “searching” the same meaning as “comparing.”

3. The Board’s Construction of Searching is Consistent With How Searching is Used in the ‘717 Patent

Microsoft argues that the Board determined the meaning of “searching” “in a non-contextual way.” But the Board’s construction of “searching” is consistent with the multiple uses of the word searching in the specification, particularly where searching is disclosed as occurring in multiple locations. *See*, e.g., A00083at 8:50-53. It is Microsoft’s attempt to limit “searching” to “comparing” that does violence to the context of the ‘717 Patent. Construing “searching” to meaning “comparing” makes no sense in the context of the embodiments that search in multiple locations. Furthermore, construing searching to mean “comparing” neglects the distinctive use of the words “searching” and “comparing” in the ‘717 Patent.

4. There is No Indication that the Board Did Not Properly Consider the Specification

Microsoft argues that the Board erroneously limited the role of the specification to be a source of explicit definitions. While the Board did note the absence of an express definition of "searching" in the specification, there is no indication that this ended the Board's consideration of the specification. The Board expressly stated that claim language is given the broadest reasonable interpretation "in light of the specification of the patent." A00010. The Board noted uses of the specification in the parties' arguments (*see*, e.g., A00011, A00012, A00013, A00015, A00017, A00019) and expressly used the specification in construing other terms (*see*, e.g., A00012, A00013, A00017). Having acknowledged the role of the specification in claim construction, noting the parties' use of the specification, and using the specification several times itself in construing claim terms, there is no reason to believe that the Board lost sight of the proper role of the specification in claim construction.

It is Microsoft, proposing that the word "searching" means "comparing" – a word used in distinctly different embodiments in the specification -- that appears to have lost sight of the proper role of the specification in claim construction.

5. The Board Did Not Erroneously Read Limitations into the Claims

In finding that the word "searching" included searching "a set of potentially many data objects" the Board did not read any limitation into the claims, but

instead gave "search" its common ordinary meaning, as evidenced by the dictionary definitions it cites. A00017. Microsoft's assertion that this is not mentioned in the specification is simply false. As pointed out above, the specification clearly contemplates and describes searching multiple locations which necessarily means searching "potentially many data items." It is Microsoft's attempt to limit searching to a single comparison, that is unreasonable and improperly limits the claims. It is improper because as Proxyconn has pointed out there are separate embodiments disclosed in the '717 Patent that describe comparing, and that use the word "comparing" to do so. It is further improper, because limiting searching to comparing two values makes no sense in the context of the embodiments where multiple locations are searched.

Microsoft tries to pin the meaning of searching on Fig. 5 which uses the term "check for." However, Fig. 5 refers to the operation of the system of Fig. 4, *see* A00082 at 5:49-51, which expressly includes a "comparing means." Microsoft is notably silent about Figs. 8 and 12, which show "search for digest" and Fig. 15 which shows "look for digest." Different words have different meanings, and limiting the meaning of one word used to describe some embodiments with the meaning of a different word, used to describe different embodiments is improper. It is Microsoft, not the Board that is improperly reading limitations into the specification.

6. The Board's Construction is not Unclear

Microsoft complains that the Board's construction of "searching" is unclear. There is nothing unclear about construing "searching for data with the same digital digest in said network cache memory" as requiring "an ability to identify a particular data object with the same digital digest from a set of potentially many data objects stored in the network cache memory." Searching means seeking a particular data object out of potentially many data objects residing in the cache memory. Microsoft suggests that the Board somehow mixed up method and apparatus elements, but there is no indication that the Board was requiring any particular device or apparatus rather than explaining what it found was the broadest reasonable interpretation of "searching."

The Board said that "searching for data with the same digital digest in said network cache memory" was identifying a particular data object with the same digital digest from a set of potentially many data objects stored in the network cache memory. Microsoft says that "searching for data with the same digital digest in said network cache memory" is comparing two digests. Microsoft's construction is neither the broadest, nor is it reasonable. The Board's construction, on the other hand, is informed by the plain and ordinary meaning of the word both in general, and in the specific field. It is consistent embodiments of the specification that employ the word "searching", and in particular those embodiments which involve

searching in multiple locations, and it accounts for the different uses of "searching" and "comparing" in the specification.

CONCLUSION

For all the reasons given, Proxyconn respectfully requests that: (1) the erroneous construction of Sender/Computer and Receiver/Computer be corrected, applying the claim construction principles of *Phillips* or at least restricting the construction to the broadest *reasonable* construction, thereby requiring the reversal of the determination that Santos anticipates Claims 1, 3, 10, 22, and 23. (2) the erroneous construction of *"Gateway . . . Between at Least Two Other Computers"* be corrected, applying the claim construction principles of *Phillips* or at least restricting the construction to the broadest *reasonable* construction, thereby requiring the reversal of the determination that DRP anticipates Claims 6, 7 and 8. (3) That the erroneous construction of sender/computer and receive/computer be corrected, applying the claim construction principles of *Phillips* or at least restricting the construction to the broadest *reasonable* construction, thereby requiring the reversal of the determination that Yohe anticipates claim 10. (4) That the Board's clearly erroneous determination that Perlman's analogous prior art be reversed, and: (5) the Board's determination that it would have been obvious, without any reason, to reconfigure Yohe and Perlman be reversed, requiring the reversal of the determination that claims 1, 3, and 10 are obvious. (6) That the determination that Santos discloses a receiver/computer with permanent storage

memory and searching in a particular location be reversed, requiring the reversal of the determination that Claims 22 and 23 are anticipated by Santos. (7) That the Board erred in finding that merely looking at data in Santos constitutes searching, requiring the reversal of the determination that in Claim 23 is unpatentable. (8) That the determination that DRP discloses receiving a response signal be reversed, requiring the reversal of the determination that DRP anticipates Claim 11, 12 and 14. (9) In the event that the grounds for unpatentability are not reversed, that the additional requirements in excess of Patent Office regulations sua sponte imposed by the Board be rejected, requiring reversal of the decision not to grant Proxyconn's Motion to Amend: and (10) that the Board's construction of "searching" be affirmed, there by affirming the Board's determination that the Petition did not establish claim 24 was unpatentable.

Respectfully submitted.

/s/ Matthew L. Cutler
Matthew L. Cutler
Bryan K. Wheelock
HARNESS DICKEY & PIERCE, P.L.C.
7700 Bonhomme, Suite 400
St. Louis, MO 63105
(314) 726-7500
mcutler@hdp.com
bwheelock@hdp.com

Attorneys for Cross-Appellant Proxyconn, Inc.

September 22, 2014

ADDENDUM

Trials@uspto.gov
Tel: 571-272-7822

Paper 73
Entered: February 19, 2014

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION
Petitioner

v.

PROXYCONN, INC.
Patent Owner

Case IPR2012-00026
Case IPR2013-00109
Patent 6,757,717

Before SALLY C. MEDLEY, THOMAS L. GIANNETTI, and MITCHELL G. WEATHERLY *Administrative Patent Judges.*

WEATHERLY, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
35 U.S.C. § 318(a) and 37 C.F.R. § 42.73

I. BACKGROUND

A. Introduction

On September 18, 2012, Microsoft Corporation ("Microsoft"), filed a petition under 35 U.S.C. §§ 311-319 for *inter partes* review of claims 1, 3, 10–12, 14, and 22–24 of U.S. Patent No. 6,757,717 ("the '717 Patent"). IPR2012-00026, Paper 6 ("the '026 Petition"). We granted the '026 Petition as to certain challenges to the patentability of claims 1, 3, 10, and 22–24, and denied the '026 Petition as to all challenges to the patentability of claims 11, 12, and 14 on December 21, 2012. IPR2012-00026, Paper 17 ("the '026 Decision").

Soon afterward, on January 11, 2013, Microsoft filed a second petition for *inter partes* review, this time challenging the patentability of claims 6, 7, 9, 11, 12, and 14 of the '717 Patent. IPR2013-00109, Paper 1 ("the '109 Petition"). Microsoft concurrently filed a motion to join IPR2013-00109 with IPR2012-00026. IPR2013-00109, Paper 7. We granted the '109 Petition as to certain challenges to patentability of claims 6, 7, 9, 11, 12, and 14 of the '717 Patent. IPR2013-00109, Paper 14 ("the '109 Decision"). We also granted Microsoft's motion for joinder and joined IPR2013-00109 with IPR2012-00026. IPR2013-00109, Paper 15.

After institution and joinder of both trials, Proxyconn, Inc. ("Proxyconn") filed its Corrected Patent Owner's Response ("Resp."). Paper 45.[1] Proxyconn also filed Patent Owner's Corrected Motion to Amend ("Mot. Amend") in which Proxyconn moved to substitute claims 35–

[1] This reference to "Paper" and all other references to "Paper" from this point forward in this Final Written Decision are to papers filed in the joined proceeding, which is captioned as IPR2012-00026 and IPR2013-00109.

41 for claims 1, 3, 6, 10, 11, 22, and 23, respectively, if the Board were to cancel any of those challenged claims as unpatentable. Paper 44.[2] This Final Written Decision addresses challenges to the patentability of claims 1, 3, 6, 7, 9–12, 14, and 22–24. Because claims 1, 3, 6, 10, 11, 22, and 23 are found unpatentable, this Decision also addresses the patentability of proposed substitute claims 35–41.

B. The '717 Patent

The '717 Patent describes a system for data access in a packet switched network. Ex. 1002, Abstract. The system has a sender/computer including an operating unit, a first memory, a permanent storage memory, and a processor. The system also has a remote receiver/computer including an operating unit, a first memory, a permanent storage memory, and a processor. The sender/computer and receiver/computer communicate through the network. *Id*. The sender/computer further includes a device for calculating digital digests on data; the receiver/computer further includes a network cache memory and a device for calculating digital digests on data in the network cache memory; and the receiver/computer and/or the sender/computer includes a device for comparison between digital digests. *Id.*

As described in the '026 Petition, the '717 Patent provides a way to reduce the amount of redundant data transmitted over a network. '026 Petition, 4. The processes described in the '717 Patent check for the identity

[2] Proxyconn filed Patent Owner's Motion to Amend under 37 C.F.R. § 42.121 on May 21, 2013. Paper 37. In an Order entered June 20, 2013, Proxyconn was granted permission to file its Corrected Motion to Amend to address typographical errors and file corrected exhibits. Paper 43. Proxyconn filed its Corrected Motion to Amend later that same day.

between two sets of data by comparing respective digital fingerprints of that data. *Id.* As described in the Summary of the Invention:

> If a sender/computer in the network is required to send data to another receiver/computer, and the receiver/computer has data with the same digital digest as that of the data to be sent, it can be assumed with sufficient probability for most practical applications that the receiver/computer has data which is exactly the same as the data being sent. Then, the receiver/computer can use the data immediately without its actual transfer through the network. In the present invention, this idea is used in a variety of ways.

Ex. 1002, col. 2, ll. 16-24.

The patent discloses several embodiments. In one, a sender/computer required to send data to a receiver/computer initially sends a digital digest of the data. If the receiver/computer already has data with the same digital digest, it uses this data as if it were actually transmitted from the sender/computer. *Id*. at col. 2, ll. 26-31. This embodiment is illustrated in Figures 5-7. Figure 5 is reproduced below:

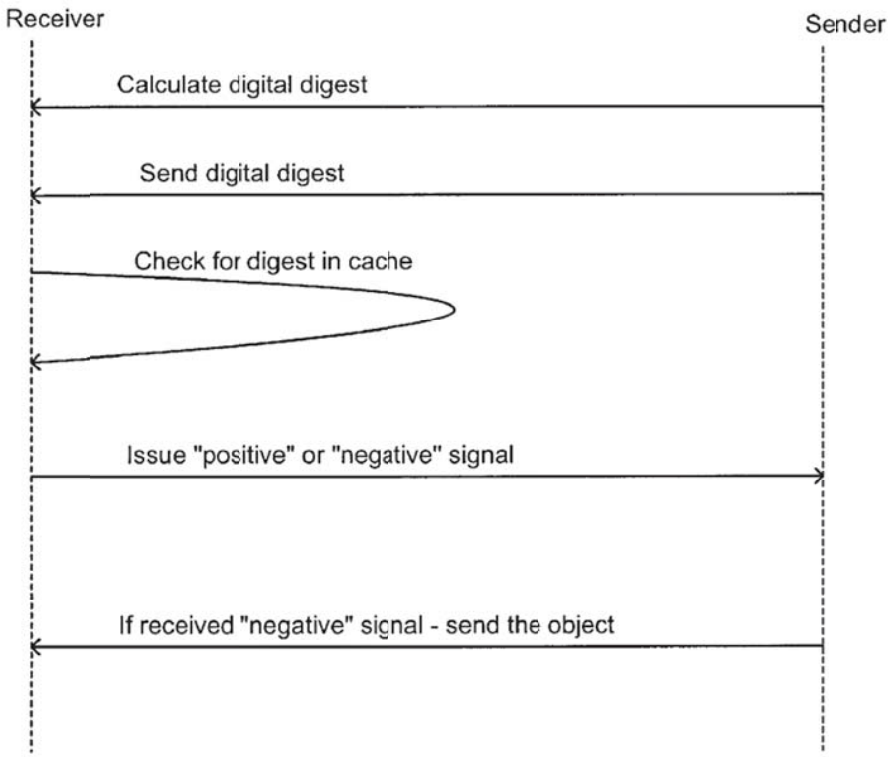


FIG. 5

Figure 5 is a schematic representation illustrating the interaction between a sender/computer and a receiver/computer according to the teachings of one embodiment of the '717 Patent. *Id*. at col. 5, ll. 49-51.

In this embodiment, the receiver/computer receives a digital digest from a sender/computer and searches its network cache memory for data with the same digest. If the receiver/computer finds such data, it uses that data as if the data were received from the sender/computer and issues a positive indication signal to the sender/computer. Otherwise it sends a negative indication signal to the sender/computer. *Id*. at col. 7, ll. 51–60.

In another embodiment, auxiliary digital digests for other data objects can be sent together with the principal digest. If the receiver/computer cannot find data having the principal digest, it searches for data with one of the auxiliary digests. If such data is found, the sender/computer is required

to send only the difference between the requested data object and the data object corresponding to the auxiliary digest. *Id*. at col. 2, ll. 31-37. The expression in the Specification "difference between the first data or data object and the second data or data object" means any bit sequence that enables the restoration of the first data, given the second data, the bit sequence, and the method employed in calculating the difference. *Id*. at col. 2, ll. 38–42. This embodiment is illustrated in Figures 8-10. Figure 8 is reproduced below:

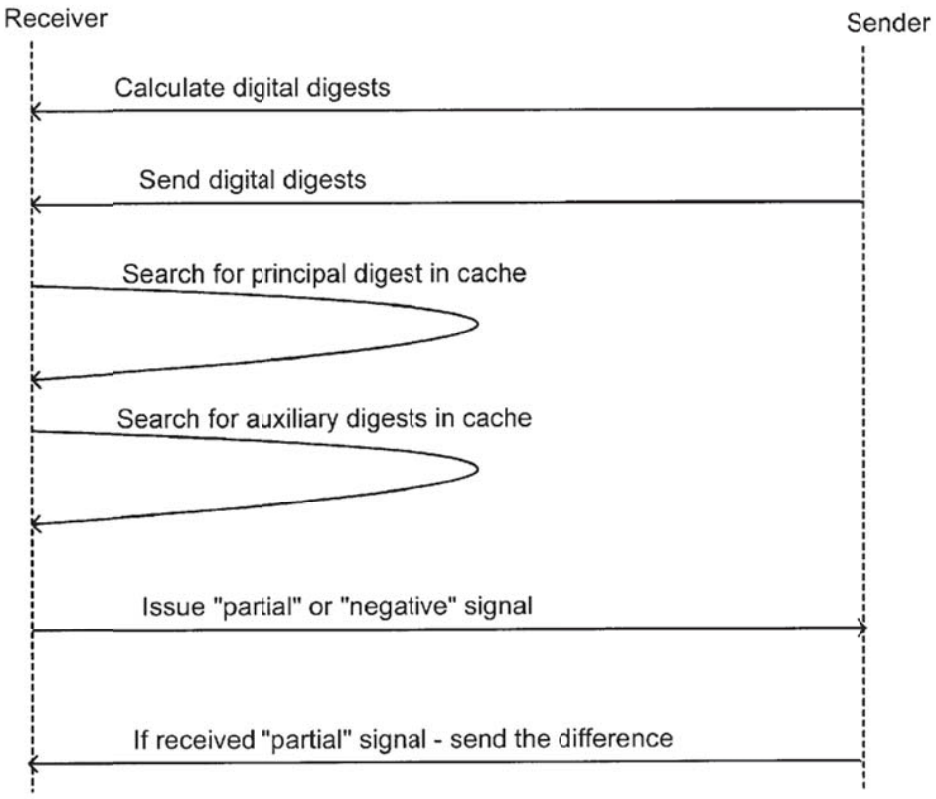


FIG. 8

Figure 8 is a schematic representation illustrating the interaction between a sender/computer and a receiver/computer according to the teachings of another embodiment of the invention. *Id*. at col. 5, ll. 59-61.

In this embodiment the sender/computer sends the principal and auxiliary (e.g., of a previous version of the data requested) digests to the receiver/computer. Upon receiving a message with these digital digests from the sender/computer, the receiver/computer searches its network cache memory for data having the same principal digest. If such data is found, the receiver/computer uses the data as if the data were received from the sender/computer and issues a positive indication signal to the sender/computer. Otherwise, the receiver/computer searches its network cache memory for data with the auxiliary digests. If it finds data with a digital digest substantially equal to one of the auxiliary digests, it issues a partial indication signal to the sender/computer, along with a reference to the digest. Otherwise it issues a negative indication signal to the sender/computer. *Id*. at col. 8, ll. 11-39.

C. Exemplary Claims

Claims 1, 6, 10, 11, and 22 are the independent claims among the challenged claims of the '717 Patent. Claims 1, 6, and 10 are directed to systems, and claims 11 and 22 are directed to methods. The independent challenged claims, which are illustrative of the claims at issue in this *inter partes* review, recite:

> 1. A system for data access in a packet-switched network, comprising:
>
> a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through said network;

said sender/computer further including means for creating digital digests on data;

said receiver/computer further including a network cache memory and means for creating digital digests on data in said network cache memory; and

said receiver/computer including means for comparison between digital digests.

6. A system for data access in a packet-switched network, comprising:

a gateway including an operating unit, a memory and a processor connected to said packet-switched network in such a way that network packets sent between at least two other computers pass through it;

a caching computer connected to said gateway through a fast local network, wherein said caching computer includes an operating unit, a first memory, a permanent storage memory and a processor;

said caching computer further including a network cache memory in its permanent storage memory, means for calculating a digital digest and means for comparison between a digital digest on data in its network cache memory and a digital digest received from said packet-switched network through said gateway.

10. A system for data access in a packet-switched network, comprising:

a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through a network;

said sender/computer further including means for creating digital digests on data, and

said receiver/computer further including a network cache memory, means for storing a digital digest received from said network in its permanent storage memory and means for comparison between digital digests.

11. A method performed by a sender/computer in a packet-switched network for increasing data access, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit data to a receiver/computer, the method comprising the steps of:

creating and transmitting a digital digest of said data from said sender/computer to said receiver/computer;

receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and

if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer.

22. A method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of:

receiving a message containing a digital digest from said network;

searching for data with the same digital digest in said network cache memory,

if data having the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting it back through said network; and

creating a digital digest for data received from said network cache memory.

Ex. 1002, col. 10, l. 31 to col. 12, l. 45.

D. Remaining Challenges to the Patentability of Claims

We instituted this *inter partes* review in connection with the following challenges to the patentability of claims in the '717 Patent:[3]

1. Anticipation by Perlman: claims 1, 3, and 22-24;
2. Anticipation by Yohe: claims 1, 3, 6, 7, 10, 22, and 23;
3. Anticipation by Santos: claims 1, 3, 10, 22, and 23;
4. Anticipation by DRP: claims 6, 7, 9, 11, 12, and 14;
5. Obviousness over the combination of Perlman and Yohe: claims 1, 3, 10, and 22-24; and
6. Obviousness over the combination of Mattis and DRP: claims 6, 7, 9, 11, 12, and 14.

'026 Decision 25–26; '109 Decision 20.

II. ANALYSIS

A. Claim Interpretation

We interpret patent claim language in an *inter partes* review by ascribing to that language its broadest reasonable meaning in light of the specification of the patent. 37 C.F.R. § 42.100(b); Office Patent Trial

[3] The challenges to patentability are based upon five prior art references: US 5,742,820, issued Apr. 21, 1998 (Ex. 1003) ("Perlman"); US 5,835,943, issued Nov. 10, 1998 (Ex. 1005) ("Yohe"); Santos and Wetherall, INCREASING EFFECTIVE LINK BANDWIDTH BY SUPPRESSING REPLICATED DATA (June 1998) (Ex. 1004) ("Santos"); THE HTTP DISTRIBUTION AND REPLICATION PROTOCOL, W3C Note (August 25, 1997), retrieved from http://www.www3.org/TR/NOTE-drp-19970825 (IPR2013-00109, Ex. 1003) ("DRP"); US 6,292,880 B1, issued Sep. 18, 2001 (IPR2013-00109, Ex. 1004) ("Mattis").

Practice Guide, 77 Fed. Reg. 48,756, 48,766 (Aug. 14, 2012). We also interpret claim language according to its ordinary and customary meaning to one of ordinary skill in the art in the context of the entire disclosure. *In re Translogic Tech., Inc.*, 504 F.3d 1249, 1257 (Fed. Cir. 2007).

We expressly interpret below only those claim terms that require analysis to resolve arguments related to the patentability of the challenged claims. Except as otherwise stated, we interpret the remaining claim terms as set forth in the '026 Decision and the '109 Decision.

1. Data Access

Each contested claim recites "data access." Ex. 1002, col. 10, l. 31 (claims 1, 3), col. 10, l. 64 (claims 6, 7, and 9), col. 11, l. 20 (claim 10), col. 11, l. 35 (claims 11, 12, and 14), col. 12, l. 30 (claims 22–24). Proxyconn urges that "data access" means "obtaining data . . . on a remote computer on a network, in response to a request from a client." Resp. 11 (citing Ex. 1002, col. 1, ll. 18–26; *id. at* col. 7, ll. 65–67). In support, Proxyconn cites portions of the Specification of the '717 Patent that describe exemplary data transmission sessions in which a network client "requests" data from a server. The first cited portion describes such interactions between a client and server as "prior art." Ex. 1002, col. 1, ll. 18–26. The second cited portion states: "This transaction begins with a receiver/computer sending a request to the sender/computer." *Id*. at col. 7, ll. 65–67. The phrase "[t]his transaction" refers to the interaction between the receiver/computer and the sender computer depicted in Figures 5–7 of the '717 Patent. *Id*. at col. 7, ll. 51–67.

By contrast, Microsoft contends that "data access" means "data acquisition." Microsoft Corporation's Reply to Patent Owner's Corrected

Response ("MS Reply"), 2 (Paper 46). Microsoft dismisses the portions of the Specification that Proxyconn cites as neither mentioning "data access" nor narrowly defining "data access." *Id*. Microsoft argues that other portions of the Specification imply that the step of the receiver/computer requesting data is merely optional. *Id*. (citing Ex. 1002, 8:37–39). Proxyconn's expert, Dr. Konchitsky, testified that the Specification describes scenarios in which a sender transmits data to a receiver without a request from the receiver. *See* Ex. 1024, 69:1–24, 71:8–22 (describing the data communication method illustrated in Figure 8 of the '717 Patent). Microsoft also points out that claim 32, which is not challenged, explicitly recites a method in which a client sends a request for data to a server. Paper 72, Final Hearing Transcript 10:9-12, 79:22–80:9 ("Tr.").

Both parties' interpretations of "data access" are too narrow. Neither the challenged claims nor the Specification expressly limits "data access" to require a "request from the client" as proposed by Proxyconn. The claims merely recite "data access." Even though the Specification describes examples in which the client requests data from a server, the Specification does not require that the client request data in all described embodiments of the claimed systems and methods. For example, the Specification expressly describes an embodiment in which "a sender/computer required to send data to a receiver/computer . . . initially sends a digital digest of the data." Ex. 1002, col. 2, ll. 26–28. "[L]imitations are not to be read into the claims from the specification." *In re Van Geuns*, 988 F.2d 1181, 1184 (Fed. Cir. 1993) (citing *In re Zletz*, 893 F.2d 319, 321(Fed. Cir. 1989)). We decline to do so here.

Microsoft's position is similarly unsupported by the claims themselves or the Specification. None of the challenged independent claims affirmatively recites that the receiver/computer *acquires* data from the sender/computer. Microsoft cites no portion of the Specification, and we find no support for the proposition that the Specification equates "data access" with "data acquisition."

We determine that the plain meaning of "data access" is clear. Independent challenged claims 1, 6, 10, 11, and 22 recite "access" as a noun modified by "data." Ex. 1002, col. 10, l. 31 (claims 1, 3), col. 10, l. 64 (claims 6, 7, and 9), col. 11, l. 20 (claim 10), col. 11, l. 35 (claims 11, 12, and 14), col. 12, l. 30 (claims 22–24). "Access" plainly means the "freedom or ability to obtain or make use of." MERRIAM WEBSTER'S COLLEGIATE DICTIONARY 6 (10th ed. 1999). We conclude, therefore, that the claimed systems and methods recite "data access" to refer to the freedom or ability to obtain or use data. Although obtaining or acquiring data requires access to that data, access to the data need not involve acquisition of that data.

2. Permanent Storage Memory

Claims 1, 3, 6, 7, 9, 10, and 22–24 recite "permanent storage memory." Ex. 1002, col. 10, l. 31 – col. 13, l. 8. Proxyconn argues that "permanent storage memory" means non-volatile memory that can be used for writing and reading data and does not refer to read-only memory ("ROM"). Resp. 12. The Specification states "an example of a permanent storage memory may be a disk drive, a flash RAM or a bubble memory." Ex. 1002, col. 7, ll. 38–40. In support of its proffered definition of "permanent storage memory," Proxyconn also cites Yohe's statement that "'[p]ermanent storage memory,' as used herein, includes but is not limited

to, disk drive, flash RAM or bubble memory, for example." Resp. 12 (quoting Ex. 1005, col. 3, ll. 5–7).

Microsoft counters that "permanent storage memory" is not restricted to non-volatile memory that permits multiple write operations, but may also include storage that is write-once, read-many ("WORM") memory. MS Reply 2. Microsoft contends that a CD optical storage disc, a type of non-volatile WORM memory, would constitute permanent storage memory. *See id*. (citing Ex. 1024, 88:7–89:12). Thus, the dispute centers on whether "permanent storage memory" encompasses ROM and other types of WORM types of non-volatile memory.

The testimony of both experts persuades us that a skilled artisan would interpret "permanent storage memory" to cover non-volatile memory that supports multiple write operations. Dr. Long equated the "permanent storage" described in the '717 Patent with a "disk" or "flash" memory. Ex. 1026, 97:15–98:10. Dr. Konchitsky testified that a skilled artisan would have considered "permanent storage memory," which enables writing or storing of information, to differ from "permanent memory," which can only be read after being written one time "in factory." Ex. 2002 ¶ 21. The ability to write data many times to permanent storage memory is consistent with the way that "permanent storage memory" is used in the context of at least claim 6. Claim 6 recites a "caching computer further including a network cache memory in its permanent storage memory." The presence of cache memory, which is likely to be written many times, in the "permanent storage memory" implies a capability to write data many times to the claimed "permanent storage memory." Because claim terms are normally used consistently throughout the patent, the usage of a term in one claim may

illuminate the meaning of the same term in other claims. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1314 (Fed. Cir. 2005). Therefore, we interpret "permanent storage memory" to mean any non-volatile memory that supports multiple write operations.

3. Sender/Computer and Receiver/Computer

Challenged claims 1, 3, 10, 11, 12, 14, and 22–24 recite either a "sender/computer" or "receiver/computer" or both. Ex. 1002, col. 10, l. 31 – col. 13, l. 8. Previously, we interpreted "sender/computer" to mean a computer that sends data and "receiver/computer" to mean a computer that receives data. '026 Decision 14. We also concluded that each of these respective computers can encompass multiple devices including intermediaries. *Id*.

Proxyconn argues that our interpretation is "inconsistent with the '717 Patent, is not the broadest *reasonable* interpretation of the claim terms, and should be revised to exclude separate intermediate computers such as gateways, proxies, routers, and caching computers." Resp. 13. Proxyconn contends that the Specification consistently refers to the sender and receiver computers as separate devices.

Microsoft contends that we correctly interpreted the computers to encompass multiple devices including intermediate devices. The Specification represents the receiver and sender computers (46, 42 respectively) in decidedly schematic form, as shown in Figures 4, 11, and 14, reproduced below.

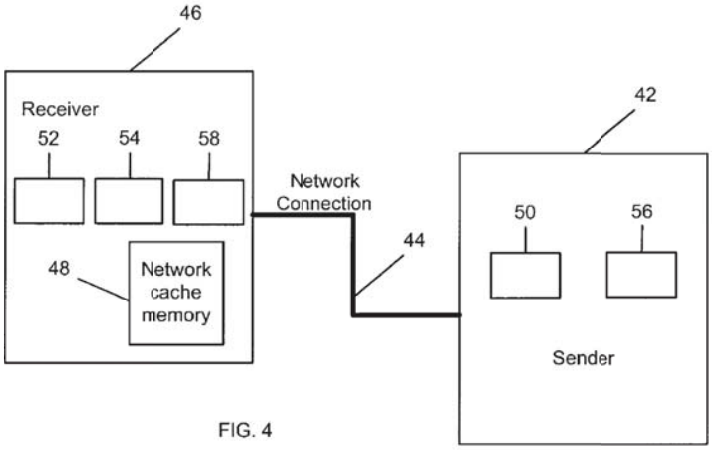


FIG. 4

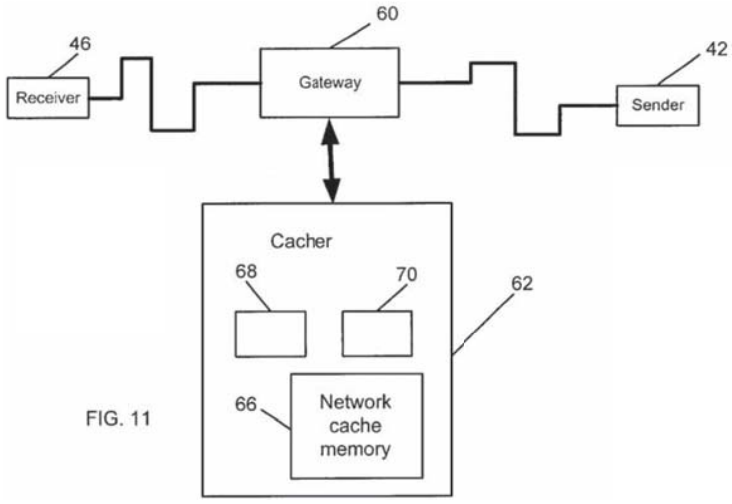


FIG. 11

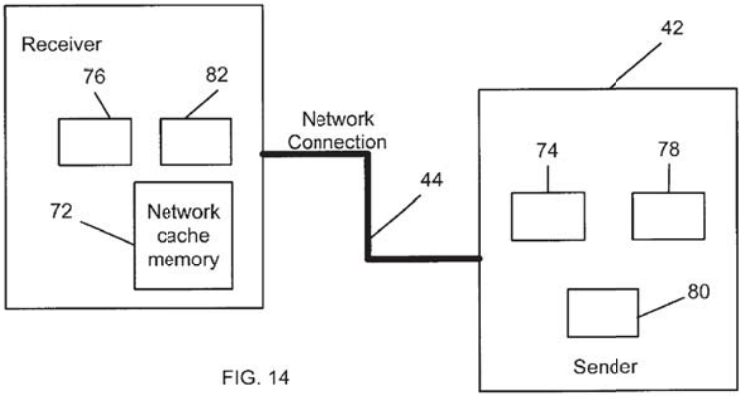


FIG. 14

Figures 4, 11, and 14, reproduced from top to bottom above, schematically illustrate the claimed receiver/computer and sender/computer in various network configurations as functional block diagrams.

Figure 4 illustrates receiver/computer 46 as a collection of functionally defined subsystems 48, 52, and 54, which are described as follows: "[T]he receiver/computer has calculating means 52 for calculating a digital digest on data stored in its network cache memory 48. The receiver/computer also has comparison means 54 for comparing between such a calculated digital digest and a digital digest received from the network." Ex. 1002, col. 7, ll. 32–37.

The Specification appears to have one instance in which a computer is described as being separate from or integral with another computer. The Specification implies that gateway 60 and caching computer 62 may be separate devices, but only by noting that "gateway computer **60** may be integrally formed with the caching computer." Ex. 1002, col. 9, ll. 6–8. The Specification, along with the above figures, conveys to a skilled artisan that the described computers, including the receiver/computer and the sender/computer, may or may not be located in separate housings. Accordingly, Proxyconn has not persuaded us to modify the original interpretation of "receiver/computer" and "sender/computer."

4. Gateway . . . Between at Least Two Other Computers

Independent claims 6 recites a "gateway . . . connected to said packet-switched network in such a way that network packets sent between at least two other computers pass through it." Ex. 1002, col. 10, l. 66 – col. 11, l. 2. Claims 7 and 9, which depend upon claim 6, also include the "gateway" and "two other computers." *Id*. at col. 11, ll. 13–20. Proxyconn asserts that "two other computers" refers to "the sender/computer and the receiver/computer." Resp. 15 (citing Ex. 1002, col. 2, ll. 44–47). The cited portion of the Specification, however, merely recites verbatim the language

of claim 6 relating to the gateway and two other computers. Therefore, the cited portion has not been shown to support Proxyconn's contention. Microsoft contends that no such limitation exists on the "two other computers" and that these computers may be *any* two other computers connected on the network to the gateway. *See* '109 Petition 13–14, Appendix A 5–6.

We agree with Microsoft. Claim 6 plainly and unambiguously recites "two other computers" as a limitation on the manner in which the "gateway" is "connected to said packet switched network." That is, the gateway is connected to the network so that "network packets sent between at least two other computers pass through it [i.e., the gateway]." Applying the broadest reasonable interpretation, we conclude that claim 6 does not limit which computers may constitute the "two other computers" between which the gateway is connected.

5. Means for comparison between digital digests

a. Claims 1 and 3

Claim 1 recites "means for comparison between digital digests." Resolution of the parties' arguments relating to whether Yohe anticipates claims 1, 3, and 10 requires that we interpret "digital digests" as recited in the comparison means. We interpret "digital digests" by reading claim 1 in its entirety. Claim 1 recites that both the sender and receiver include "means for creating digital digests on data." We conclude that the "digital digests" recited in the means for comparison refers to the "digital digests on data" that are recited earlier in claim 1 in the "means for creating."

b. Claim 10

Claim 10, like claim 1, recites that the receiver includes "means for comparison between digital digests." Also like claim 1, claim 10 recites that the sender includes a "means for creating digital digests on data." By contrast to claim 1, claim 10 does not recite a "means for creating" in the receiver. Ex. 1002, col. 11, ll. 20–33. Instead, the receiver includes a "means for storing *a digital digest* received from said network." *Id.* at col. 11, ll. 31–32 (emphasis added). The reference to "a digital digest" rather than "the digital digest on data" in the storing means implies that the receiver can store any type of digital digest received from the network.

Therefore, the "digests" that are compared in the "means for comparison" recited in claim 10 need not be the two digests on data created by the sender and receiver. Instead, the "means for comparison between digital digests" recited in claim 10 refers to structure that can compare any digital digest received from the network with any other digital digest.

6. Searching for Data with the Same Digital Digest

Claims 22–24 recite a step of "searching for data with the same digital digest." Proxyconn argues that the "searching" step requires the capability to identify particular data "with the same digital digest" from among a set of data that potentially contains multiple items. *See* Resp. 6, 20–21 (attempting to distinguish claims 22–24 from Perlman), 27–28 (attempting to distinguish claims 22 and 23 from Yohe), 35–36 (attempting to distinguish claim 23 from Santos). Microsoft contends that the '717 Patent equates "search" with "check for" and that the Specification never describes any "search" method other than "comparing two digest values for a match." MS Reply 4. Microsoft asserts that the recited step of "searching for data with the same

digital digest," merely requires comparing a digest for a data object received from the network with a digest of the receiver's copy of that data object. *Id*. at 5.

The Specification never expressly defines "search." Nonetheless, the plain meaning of "search" is: "to look into or over carefully or thoroughly in an effort to find or discover something." MERRIAM WEBSTER'S COLLEGIATE DICTIONARY 1053 (10th ed. 1999). Two dictionaries in the relevant field of computing technology define "search" as it would be understood by a skilled artisan as follows:

1. "To scan one or more data elements of a set in order to find elements that have a certain property," IBM DICTIONARY OF COMPUTING 600 (10th ed. 1993); and
2. "(information processing). To examine a set of items for those that have a desired property," IEEE STANDARD DICTIONARY OF ELECTRICAL AND ELECTRONICS TERMS 808 (3d ed. 1984).

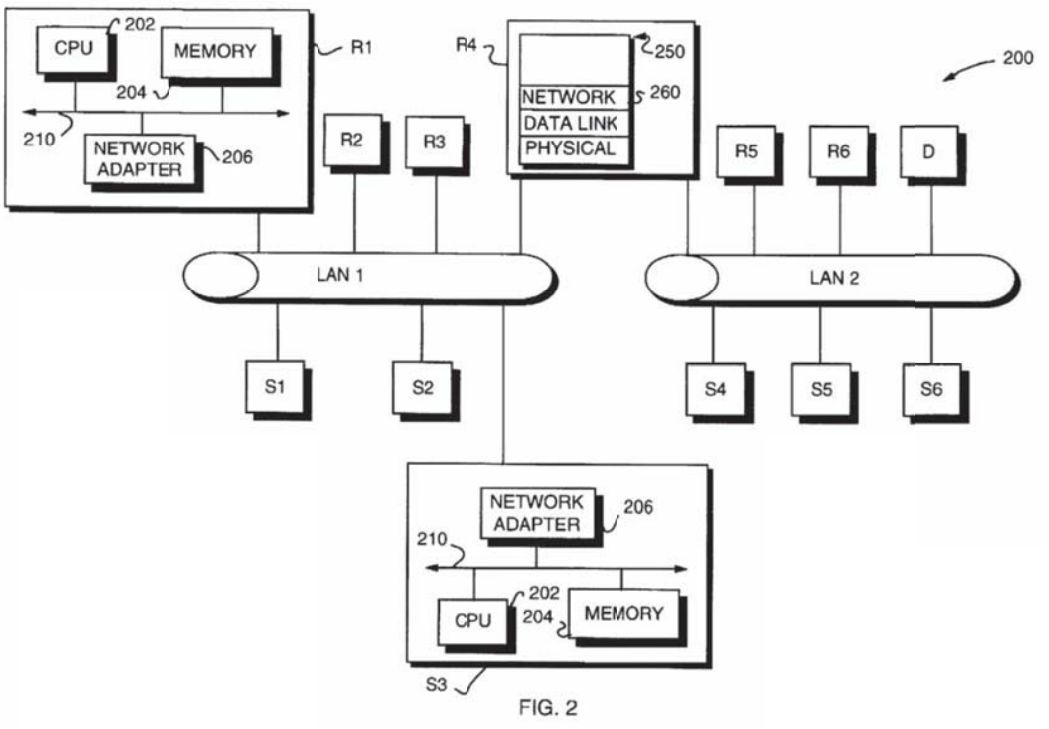
These dictionary definitions reflect that a skilled artisan would have understood "search" to involve analyzing a set of items to identify one particular item from among a set of items. A "set" refers to "a number of things of the same kind that belong or are used together," MERRIAM WEBSTER'S COLLEGIATE DICTIONARY 1071 (10th ed. 1999), or "[a] finite or infinite number of objects of any kind, of entities, or of concepts that have a given property or properties in common," IBM DICTIONARY OF COMPUTING 618 (10th ed. 1993). While a set can contain one item, a "search" for a desired member of a "set" requires a capability to examine more than one item to identify a particular item within that set. Therefore, we conclude that "searching for data with the same digital digest in said

network cache memory" requires an ability to identify a particular data object with the same digital digest from a set of potentially many data objects stored in the network cache memory.

B. The Prior Art

1. Perlman

Perlman generally relates to synchronization of information across a computer network. Ex. 1003, col. 1, ll. 6–8. Perlman's Figure 2 (reproduced below) is a block diagram of two computer networks to which multiple nodes, which include routers R1–R6, source nodes S1–S6, and a destination node D, are connected.



Perlman's Figure 2 is a block diagram of two computer networks to which multiple nodes are connected.

Perlman states that the "nodes are typically general-purpose computers" and that "[e]ach node typically comprises a . . . memory unit **204**" which may

include "storage locations typically composed of random access memory (RAM) devices." *Id*. at col. 5, ll. 40–47.

Packetized data is transmitted across the network with each packet having the address of its final destination and the address of the next node to which it will travel along the route to the final destination. *Id*. at col. 5, l. 65 – col. 6., l. 1. The final destination address remains constant, but the "next destination" address changes as the packet moves from node to node in the network. *Id*. at col. 6, ll. 1–4. Upon arrival of a packet to a router, the router determines the next destination address of the packet based on algorithms to calculate a path to the final destination. *Id*. at col. 6, ll. 5–24. For this mode of transmission to work, every router must determine and communicate its location in the network to other nodes on the network. *Id*. at col. 6, ll. 25–45. These network "maps" must be synchronized to ensure that data packets arrive at the correct final destination. *Id*. at col. 6, ll. 46–53.

Perlman synchronizes these network maps by having one designated router (e.g., R4) periodically calculate and send a digest of that map (called a "complete sequence numbers packet" or CSNP) to all other routers on the network. *Id*. at col. 6, l. 47 – col. 7, l. 55. When each of the other routers receives the CSNP digest from R4, each of those routers compares that received CSNP digest to a digest of the network topology calculated locally by the receiving router. *Id*. at col. 7, ll. 56–63.

Perlman also describes an "alternate embodiment" in which "high-level and low-level identifiers are bundled within the same hello message that is periodically broadcast by the designated router, **R4** to the other routers." *Id*. at col. 8, ll. 25–28. For this embodiment, the receiving router first compares its locally generated high-level digest with received high-

level digest 710. *Id*. at col. 8, ll. 32–35. If the two digests are not the same, then the receiving router calculates low-level digests for fragments of its database and compares each of these low-level digests with corresponding received low-level digests 725a–c. *Id*. at col. 8, ll. 36–39. Based on these comparisons, the receiving router determines which fragments of its database require updating. *Id*. at col. 8, ll. 39–42.

2. Yohe

Yohe generally describes an "apparatus for increased data access in a network [that] includes a file server computer having a permanent storage memory, [and] a cache verifying computer operably connected to the file server computer in a manner to form a network for rapidly transferring data." Ex. 1005, Abstract. Yohe's Figure 2, reproduced below, schematically illustrates the configuration of the data access network.

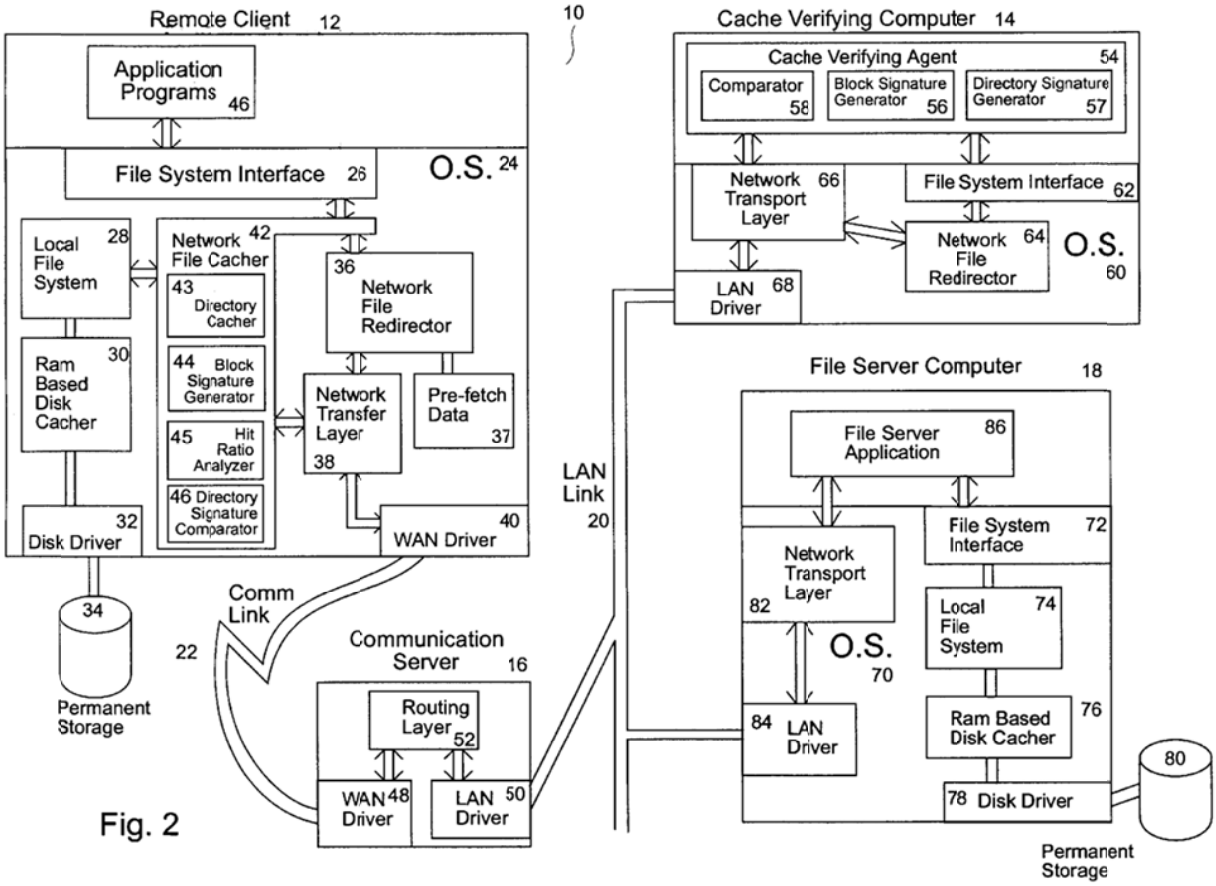


Fig. 2

Yohe's Figure 2 is a block diagram of a computer network including a remote client 12, cache verifying computer 14, communication server 16, and file server computer 18 that are connected to each other via either a LAN or WAN.

Yohe's apparatus reduces the time required for a remote client to access data on a file server using a caching computer and caching technique. *Id*. at col. 4, ll. 27–31.

Yohe's remote client computer 12 expressly includes a processor, operating system, permanent storage memory, and other memory. *Id*. at col. 2, ll. 51–54. The remote client computer also includes network file cacher 42 with block signature generator 44 and directory signature comparator 46. *Id*. at col. 5, ll. 1–6.

Similarly, Yohe's file server computer also expressly includes a processor, operating system, memory, and permanent storage memory. *Id*. at col. 3, ll. 22–24. The file server computer is "operably connected" to cache verifying computer 14, which includes "means for performing an operation on data stored in the permanent storage memory of the file server computer to produce a signature of the data characteristic of one of a file and directory." *Id*. at col. 2, ll. 43–51. Yohe describes two means for producing a signature, or digest, block signature generator 56 and directory signature generator 57. *Id*. at col. 5, ll. 14–17.

Yohe describes cache verifying computer 14 as having an operating system, a first memory, and a processor. *Id*. at col. 2, ll. 46–47. Cache verifying computer 14 incorporates cache verifying agent 54 consisting of block signature generator 56, directory signature generator 57, and comparator 58. *Id*. at col. 5, ll. 14–17. The cache verifying computer is also "operably connected" to the file server computer so that block signature generator 56 can create digests for data files stored in permanent storage 80. *Id*. at col. 2, ll. 43–51.

Yohe's client computer and the cache verifying computer thus each have capability to generate digests for data stored on the client and file server respectively. When Yohe's client computer requests data (e.g., to read a file stored in permanent storage on the file server), it generates a read request with an embedded digest and sends that request to the cache verifying computer. The cache verifying computer generates a digest for the requested file as that file exists in the permanent storage memory of the file server. The cache verifying computer then compares the two digests to determine whether the files stored on the remote client and file server are the

same. If not, the file server's version is sent to the remote client. If so, the remote client uses its locally stored version of the file. *Id*. at col. 6, ll. 22–37; figs. 6 and 7.

Yohe does not compare any particular received digest with more than one locally generated digest. Yohe describes two types of comparators for analyzing two versions of a digest. The first is comparator 58 in cache verifying computer 14, which compares digests for data files. *Id*. The second is directory signature comparator 46 in remote client 12, which compares one-by-one a series of digests for directory sub-objects that are received from the cache verifying computer with the locally generated digests for corresponding directory sub-objects. *Id.* at col. 7, l. 6 – col. 8, l. 25 (describing steps performed in DIRECTORY REQUEST function as shown in Figures 15 and 16).

3. Santos

Santos describes a compression architecture that prevents transmission of replicated data to increase bandwidth in a packet switched environment such as the Internet. Ex. 1004, 2. The bandwidth savings is achieved by transmitting repeated data as a short dictionary token, using caches of recently-seen data at both ends of the link to maintain the dictionary, and encode and decode the tokens. *Id.* at 5. The approach of Santos is based on the insight that the "fingerprint" of a data segment is an inexpensive name for the data itself, both in terms of space and time. *Id.* Santos uses the MD5 hash algorithm for his implementation, but states that other "fingerprints" could be used. *Id.* Figure 4 of Santos is reproduced below:

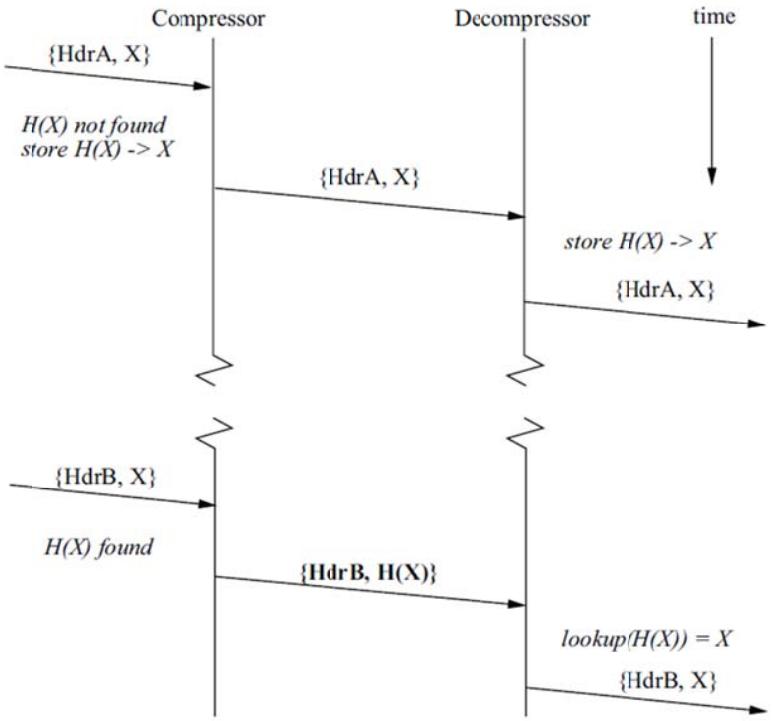


Figure 4: Compression protocol

Figure 4 of Santos shows a message exchange sequence from a sender (compressor) to a receiver (decompressor). Ex. 1004, 7.

The upper portion of the figure illustrates the sequence of events when the compressor receives a packet having header HdrA whose fingerprint H(X) is not in the cache. When this occurs, the compressor stores packet contents X in its cache, indexed by its fingerprint H(X), and forwards the header and contents across the link. The lower portion of the figure illustrates the sequence of events occurring when the compressor receives a packet having header HdrB and a fingerprint H(X) that is found in the cache. *Id*. at 7-8. When this occurs, the compressor sends the header and fingerprint, thus achieving a savings in bandwidth. *Id.* at 8. Santos implements the compressor and decompressor as two Pentium II based

machines with 128 MB of RAM and a cache of 200 MB running a Linux operating system. *Id*. at 9.

4. DRP

DRP describes a protocol for improving the efficiency and reliability of data distribution over HTTP. DRP, 2, ll. 12–13. One of the goals is to avoid downloading the same data more than once. *Id*. at 2, ll. 29–30. The protocol described in DRP makes use of content identifiers based on checksum technology. *Id*. at 2, ll. 37–39. A content identifier can be used uniquely to identify each piece of data or content and to determine whether two pieces of content are identical. An example of a checksum algorithm that can be used for this purpose is the MD5 message digest algorithm. *Id.* at 3, ll. 24–25.

To describe the exact state of a set of data files, DRP uses a data structure called an index. *Id.* at 4, l. 37. An index is a snapshot of the state of a set of files at a particular moment in time. It is typically stored in memory as a data tree structure, but to enable clients and servers to communicate this information over HTTP, an index can be described using XML. *Id*. at 4, ll. 39–42.

A DRP index is retrieved by giving a uniform resource locator ("URL") to the index. *Id*. at 5, l. 22. The index can be stored in any file and can be retrieved using a normal HTTP GET request. *Id*. at 5, ll. 22–23. Once the initial download is complete, a client can update content by downloading a new version of the index and comparing it against the previous versions of the index. Because each file entry in the index has a content identifier, the client can determine which files have changed and,

thus, determine the minimal set of files that need to be downloaded in order to bring the client up to date. *Id*. at 5, ll. 31–33.

An HTTP header field called Content-ID is used to specify the current correct version of the file that is requested by the client. The server can use the content identifier in the Content-ID field to determine if the requested version of the file can be delivered to the client. *Id*. at 7, ll. 30–32. If no content identifier is specified in the HTTP GET request, the server returns the current version of the file. *Id*. at 7, ll. 37–38. When a file is updated on the server, it will be downloaded by each of the clients that needs the new version. *Id*. at 8, ll. 3–4.

DRP notes that updates to files very often affect only small portions of the file, and it would therefore be much more efficient if the server could reply with only the parts of the file that have changed. *Id*. at 8, ll. 4–5. This is achieved using a "differential" GET request. When a file is modified, the client can issue a differential GET request for the file, which includes not only the content identifier of the desired version of the file, but also the content identifier of the current version of the file on the client. *Id*. at 8, ll. 11–13. In a differential GET request the content identifier of the file as it exists on the client is specified using the Differential–ID field in the HTTP header. *Id*. at 8, ll. 14–15. When the server receives a GET request that includes a Differential–ID field, it can look in its file cache for both versions of the requested file, using the content identifiers specified in the Content–ID field and the Differential–ID field. If both versions of the file are found, the server can compute the difference between the two files and return the difference, rather than the entire file. *Id*. at 8, ll. 25–28. If the server does not have access to the version of the file that is indicated by the differential

content identifier, it can ignore the differential content identifier and return the entire requested file. *Id*. at 8, ll. 32–33.

DRP also describes the use of proxy caching. In this application, an HTTP proxy can be made aware of the Content–ID and Differential–ID fields in HTTP requests and replies. *Id*. at 9, ll. 38–39. Because the content identifier is included in each GET request, the proxy can avoid accidentally returning the wrong version of the requested file. *Id*. at 9, ll. 39–40. The proxy can use the content identifier field to identify uniquely the content being transferred as the same content is likely to have the same content identifier, even when downloaded from multiple locations. The proxy can thus use this information to avoid multiple downloads. *Id*. at 9, ll. 43–45. The proxy can also use the Differential–ID header field to reply to differential GET requests. If both versions of the file are in the proxy's cache, the proxy can provide the differential reply. *Id*. at 10, ll. 1–2.

5. Mattis

According to Mattis, a key factor limiting the performance of the World Wide Web is the speed with which servers can supply information to clients via the Internet. Mattis, col. 1, ll. 53–55. Accordingly, client transaction time can be reduced by storing replicas of popular information objects in repositories geographically dispersed from the server. Each local repository for object replicas is generally referred to as a cache. *Id*. at col. 1, ll. 58–62. In some arrangements, the cache is located in a proxy server that is logically interposed between clients and the server. The proxy server is a "middleman gateway," acting as a server to the client, and the client to the server. *Id*. at col. 1. 66 to col. 2, l. 3. A proxy server equipped with a cache

is called a "caching proxy server," or just a "proxy cache." *Id*. at col. 2, ll. 3–5.

The proxy cache intercepts requests for resources that are directed from clients to the server. When the cache in the proxy has a replica of the requested resource that meet certain constraints, the proxy responds to the clients and serves the resources directly. *Id*. at col 2, ll. 6–11. In this arrangement, the number and volume of data transfers along the links are greatly reduced. As a result, network resources or objects are provided more rapidly to the clients. *Id*. at col. 2, ll. 11–14.

Mattis uses a "fingerprint" of the content that makes up the object itself to locate the object. *Id*. at col. 8, ll. 18–21. Specifically, the object key is a unique "fingerprint" or compressed representation of the contents of the object. A copy of the object is provided as an input to a hash function (e.g., MD5) and its output is the object key. Given a content fingerprint key, the content can easily be found in the cache. *Id*. at col. 8, ll. 23–36. In some embodiments of Mattis, for each of the objects, the cache also creates a name object key. The name key is created by applying a hash function to the name of the object. *Id*. at col. 8, ll. 55–58. Mattis recognizes that requests for objects typically identify requested objects by name, such as a URL, file system name, or network address. *Id*. at col. 9, l. 65 to col. 10, l. 4.

In one embodiment of Mattis, a lookup operation is used to determine whether a particular object identified by particular name is stored currently in the cache. *See id*. at fig. 9A. When the process is applied in the context of the World Wide Web, the name is a URL. *Id*. at col. 27, ll. 61–62. The cache converts the name of the object to a key value by passing the object

name or URL to hash function such as MD5. *Id*. at col. 27, ll. 63–67. The key is checked against a directory, and if the requested object is found, it is retrieved from storage and sent to the client. *Id*. at col. 28, ll. 10–14. If the object is not found in storage, the cache obtains a copy of the object from the appropriate server. *Id*. at col. 28, ll. 43–47.

C. Patentability of Original Claims

To prevail in its challenges to the patentability of claims, the petitioner must establish facts supporting its challenges by a preponderance of the evidence. 35 U.S.C. § 316(e); 37 C.F.R. § 42.1(d).

1. Anticipation

The Court of Appeals for the Federal Circuit summarized the analytical framework for determining whether prior art anticipates a claim as follows:

> If the claimed invention was "described in a printed publication" either before the date of invention, 35 U.S.C. § 102(a), or more than one year before the U.S. patent application was filed, 35 U.S.C. § 102(b), then that prior art anticipates the patent. Although § 102 refers to "the invention" generally, the anticipation inquiry proceeds on a claim-by-claim basis. *See Hakim v. Cannon Avent Group, PLC*, 479 F.3d 1313, 1319 (Fed. Cir. 2007). To anticipate a claim, a single prior art reference must expressly or inherently disclose each claim limitation. *Celeritas Techs., Ltd. v. Rockwell Int'l Corp.*, 150 F.3d 1354, 1361 (Fed. Cir. 1998). But disclosure of each element is not quite enough—this court has long held that "[a]nticipation requires the presence in a single prior art disclosure of all elements of a claimed invention *arranged as in the claim*." *Connell v. Sears, Roebuck & Co.*, 722 F.2d 1542, 1548 (Fed. Cir. 1983) (citing *Soundscriber Corp. v. United States*, 175 Ct.Cl. 644, 360 F.2d 954, 960 (1966) (emphasis added)).

Finisar Corp. v. DirecTV Grp., Inc., 523 F.3d 1323, 1334–35 (Fed. Cir. 2008). We must analyze prior art references as a skilled artisan would. *See Scripps Clinic & Res. Found. v. Genentech, Inc.*, 927 F.2d 1565, 1576 (Fed. Cir. 1991) (to anticipate, "[t]here must be no difference between the claimed invention and the reference disclosure, as viewed by a person of ordinary skill in the field of the invention").

For the reasons expressed below, Microsoft has demonstrated by a preponderance of evidence that Yohe anticipates claim 10; Santos anticipates claims 1, 3, 22, and 23; and DRP anticipates claims 6, 7, 9, 11, 12, and 14. Microsoft has failed to establish by a preponderance of evidence that Perlman and Yohe anticipate any of the remaining challenged claims.

a. Perlman

(1) Claims 1 and 3

Proxyconn argues that Perlman does not anticipate claims 1 and 3 because Perlman fails to disclose "permanent storage memory," but instead uses random access memory ("RAM"). Resp. 18 (citing Ex. 1003, col. 5, ll. 46–48)). Microsoft counters that Perlman describes permanent storage memory in the form of memory for storing an operating system. '026 Petition 17 (citing Ex. 1002, col. 5, ll. 41–52, fig. 2). We see no express statement in the cited portion of Perlman that its computers include any type of memory other than RAM. '026 Petition 17 (citing Ex. 1007, 12:5–17). Microsoft relies upon the Declaration of Darrell D. E. Long, Ph.D. Dr. Long opines, without explanation, that a skilled artisan would understand Perlman to disclose an "illustrative-embodiment router [that] is a 'general-purpose' computer . . . having, among other things, a hard disk or the like for storing

persistently data and application programs." Ex. 1007, 12:7–12 (citing Perlman, Ex. 1003, col. 5:41–52, fig. 2). However, Dr. Long also testified:

> Q. So your testimony is that each and every limitation of claim one is disclosed in Perlman, correct?
>
> A. We've got a computer — there may be — I'd have to go back and study Perlman. Maybe we don't mention permanent storage, whatever that means.
>
> Q. Okay.
>
> A. Okay. Although, certainly that's — I would consider that a triviality.
>
> Q. Okay.
>
> A. Okay. Digest, that's there. That's there. Comparisons are there. Maybe permanent memory is missing.

Ex. 1026, 172:10–22. When asked specifically how Perlman describes "permanent storage memory," Dr. Long responded, "I thought [Perlman] talked about flash in here. Certainly, it's something that a router can have and router[s] can and do have." Ex. 1026, 149:4–9. However, Dr. Long failed to identify how Perlman expressly discloses permanent storage memory. Based on his testimony, we conclude that a skilled artisan would understand Perlman to disclose routers that may or may not have permanent storage memory.

A finding of anticipation by inherency requires more than probabilities or possibilities. *Motorola Mobility LLC v. Int'l Trade Comm'n*, 737 F.3d 1345, 1350 (Fed. Cir. 2013). Based on the evidence discussed above, it is possible to infer that Perlman describes such permanent storage memory. However, Microsoft has not presented evidence that the computers or routers described by Perlman necessarily use permanent storage memory

as recited in claims 1 and 3. Therefore, Microsoft has failed to establish by a preponderance of evidence that Perlman describes a receiver/computer or sender/computer necessarily having "permanent storage memory." Without such evidence, we cannot conclude that Perlman anticipates claims 1 and 3.

(2) Claims 22–24

Proxyconn contends that Perlman fails to disclose the step recited in independent claim 22 of "searching for data with the same digital digest in said network cache memory." Resp. 20. Instead, Proxyconn urges that Perlman stores "one database identifier" that is merely "compared to the identifier received from the designated router." *Id.*; *see also* Ex. 2007, 13 (citing Ex. 1003, col. 8, ll. 32–42). Dr. Konchitsky testified, without citing any particular portion of Perlman, that Perlman's "receiving routers receive an identifier and each simply compares the received identifier with its existing identifier. The receiving routers are not searching for data files using the identifier as the key, or among multiple identifiers." Ex. 2002, ¶ 23.

Microsoft cites numerous passages from Perlman as meeting the recited "searching" step. '026 Petition, App'x. A, 17. All but one of the cited passages from Perlman describes the comparison of a received high-level database digest with a locally calculated high-level database identifier. The other cited passage (Ex. 1003, col. 8, ll. 22–49) describes an alternate embodiment in which received low-level digests are compared to locally generated low-level digests, if the received and locally generated high-level digests do not match. *See* part II.B.1 above (describing Perlman's use of high-level and low-level digests).

We conclude that Microsoft has failed to establish by a preponderance of evidence that Perlman meets the "searching" step recited in claim 22. Every portion of Perlman upon which Microsoft relies involves comparing one received digest to a corresponding locally calculated digest. These one-to-one comparisons cannot identify a particular data object from among a set of data objects as required by the "searching" step. Rather, the comparisons reveal whether a locally stored data object is synchronized with the corresponding remotely stored data object. Claims 23 and 24 depend from claim 22. We therefore, conclude that Perlman does not anticipate claims 22–24.

b. Yohe

(1) Claims 1, 3, and 10

Proxyconn first argues that Yohe fails to describe a sender/computer having permanent storage memory and means for creating digital digests of data. Resp. 21–23. Microsoft counters that the combination of Yohe's cache verifying computer 14 and file server 18 constitutes the claimed sender/computer. '026 Petition 16, App'x. A, 3; MS Reply 5–6; *see also* Ex. 1028, 16 (regarding annotated version of Yohe, Figure 2). We reject Proxyconn's argument because claims 1, 3, and 10 do not limit "sender/computer" to hardware residing in one housing. We have concluded that "sender/computer" may broadly encompass "multiple devices," and Proxyconn has directed us to no persuasive evidence that our interpretation is incorrect.

Proxyconn next argues that Yohe fails to describe a "receiver/computer" with means for creating digital digests on data. Resp. 23–24. Microsoft identifies Yohe's block signature generator 44 in

remote client computer 12 as the recited "means for creating digital digests on data." '026 Petition 20, App'x. A, 9; *see also* Ex. 1028, 13 (regarding annotated version of Yohe, Figure 2). We agree with Microsoft that Yohe's block signature generator 44 creates digests on data stored in Yohe's cache. Ex. 1005, col. 6, ll. 22–23.

Proxyconn also argues that Yohe fails to describe a "receiver/computer" with means for comparing received and locally generated digital digests on data. Resp. 24; *see also* Ex. 2007, 18–19 (regarding annotated version of Yohe, Figure 2). Proxyconn contends that comparator 58, which is part of cache verifying computer 14 and not remote client 12, compares the digests generated by block signature generators 44 and 56. Ex. 2007, 19 (annotating Yohe's Figure 2); *see also* Tr. 55:1–8 (identifying Ex. 1005, Yohe, col. 6, ll. 22 to col. 7, ll. 16 as confirming annotations on Yohe's Figure 2). Microsoft identifies Yohe's directory signature comparator 46 in remote client computer 12 as the recited "means for comparison between digital digests" that compares the digital digests on data that were separately created by the sender and receiver computers. '026 Petition 21, App'x. A, 10; *see also* Ex. 1028, 13 (regarding annotated version of Yohe, Figure 2).

We agree with Proxyconn that Yohe fails to describe a "receiver/computer including means for comparison of digital digests" as recited in claims 1 and 3. The two versions of digests on data that Yohe generates via block signature generators 44 and 56 are compared by comparator 58, which is not a component of remote client 12. For this reason, we conclude that Microsoft has not established by a preponderance of evidence that Yohe anticipates claims 1 and 3.

However, claim 10 does not require that the "means for comparison between digital digests" refers to comparing a digest generated by the receiver to a digest generated by the sender. *See* Part II.A.5.b above. Instead, we have concluded that the "means for comparison between digital digests" recited in claim 10 refers to structure that can compare *any* digital digest received from the network with another digital digest. Yohe's remote client 12 includes directory signature comparator 46 that compares a directory signature received from the cache verifying agent 54 (i.e., a portion of a sender/computer), with a directory signature retrieved from the remote client's cache. Ex. 1005, col. 8, ll. 5–11. Therefore, we agree with Microsoft that Yohe's directory signature comparator 46 is a "means for comparison between digital digests" as recited in claim 10. We determine that Microsoft has established by a preponderance of evidence that Yohe anticipates claim 10.

(2) Claims 6 and 7

Proxyconn argues that Yohe fails to describe a caching computer having network cache memory in its permanent storage memory as required in claims 6 and 7. Microsoft contends that permanent storage device 80, shown as part of Yohe's file server 18, constitutes the "permanent storage memory" of the claimed caching computer. '109 Petition 17, App'x. A,10. Essentially, Microsoft argues that the combination of Yohe's cache verifying computer 14 and file server 18 meet all the recited limitations for the caching computer of claim 6. Proxyconn contends that even if we were to agree that cache verifying computer 14, file server 18, and permanent storage device 80 were the claimed "caching computer," this combination still fails to describe network cache memory in the permanent storage

memory. Resp. 26–27. Microsoft fails to address this argument in its reply. *See* MS Reply 9. Our review of Yohe fails to identify any description of a network cache memory in permanent storage device 80. For this reason, we conclude that Microsoft has not established by a preponderance of evidence that Yohe anticipates claim 6 or its dependent claim 7.

(3) Claims 22 and 23

Proxyconn argues that Yohe fails to disclose the step recited in independent claim 22 of "searching for data with the same digital digest in said network cache memory." Resp. 27–28. Proxyconn contends that the only analysis of digests performed by Yohe's receiver (i.e., the remote client) is performed by Yohe's directory signature comparator 46, which compares one directory digest with a digest received from the cache verifying computer. *Id*. at 27. Proxyconn argues that such one-to-one comparisons are not "searching" as recited in claim 22. *Id*. at 27–28.

Microsoft does not dispute Proxyconn's characterization of the manner in which Yohe's directory signature comparator operates. MS Reply 11. Rather, Microsoft contends that Proxyconn's position is predicated on a flawed interpretation of "search" that requires more than a "single-comparison search." *Id*.

Based on our review of Yohe, the directory signature comparator 46 performs, at most, a series of one-to-one comparisons of received digests to locally generated digests for directory sub-objects. Ex. 1005, col. 7, l. 6 – col. 8, l. 25 (describing steps performed in DIRECTORY REQUEST function as shown in Figures 15 and 16). We conclude that such one-to-one comparisons do not identify a particular directory sub-object from among a set of directory sub-objects as required by the "searching" step. Rather, the

comparisons simply reveal whether a locally stored directory sub-object is synchronized with the corresponding remotely stored directory sub-object. Claim 23 depends from claim 22. We therefore conclude that Microsoft has not established by a preponderance of evidence that Yohe anticipates claims 22 and 23.

c. Santos

(1) Claims 1 and 3

Proxyconn raises two arguments allegedly distinguishing claims 1 and 3. We address each in turn below.

First, Proxyconn argues that Santos fails to describe a receiver/computer. Resp. 33–34. Proxyconn contends that Santos describes "a compressor and a decompressor, which are intermediate computers." *Id*. at 33. Microsoft counters that Proxyconn's argument rests upon an incorrect interpretation of "receiver/computer." We agree. We have interpreted "receiver/computer" to refer to "a computer that receives data." Santos's two computers on opposite ends of the communication channel can send and receive data. *See* Ex. 1004, 6 (stating that "[b]idirectional compression is achieved by using two instances of the protocol, one for each direction.") Thus, Santos uses "compressor" and "decompressor" to denote the function performed by each of two computers during transmission of specific data in a specific direction across the communication channel between them. Because data can move in both directions across that channel, both machines may function as a compressor (i.e., sender) or a decompressor (i.e., receiver). In the context of claims 1 and 3, the decompressor is acting as a "receiver/computer." We therefore reject Proxyconn's first argument.

Second, Proxyconn argues that Santos fails to describe "means for creating a digital digest on *data in the network cache memory*." Resp. 35. Proxyconn contends that because Santos's compressor and decompressor calculate the digest for data *after* the data is received but *before* the data is stored, the calculation of the digest is not on data in the network cache memory. *Id*. Proxyconn's argument rests upon an inferred interpretation of the creating means that requires a specific order of operations. More specifically, Proxyconn argues that "creating a digital digest on data in the network cache memory" implicitly requires that the receiver first read data from the network cache memory and then create a digest for that data.

Microsoft counters that claims 1 and 3 "do not require that the receiver first read the data from the cache and then and only then calculate a digest on it." MS Reply 8. Microsoft's argument also rests upon an inferred interpretation of the creating means. More specifically, Microsoft infers that the receiver's creating means need only be capable of creating a digest corresponding to data that is or will be stored in the network cache memory.

We conclude that Microsoft's interpretation is more consistent with the broadest reasonable interpretation of "means for creating digital digests on data in said network cache memory." Claim 1 and its dependent claim 3 recite systems, not methods, with hardware having recited functional capabilities. The functional language in claim 1 is not limited to a particular order of operations. "[A]pparatus claims cover what a device *is*, not what a device *does*. An invention need not *operate* differently than the prior art to be patentable, but need only *be* different." *Hewlett-Packard Co. v. Bausch & Lomb Inc.*, 909 F.2d 1464, 1468 (Fed. Cir. 1990); *see also Roberts v. Ryer*, 91 U.S. 150, 157 (1875) ("The inventor of a machine is entitled to the

benefit of all the uses to which it can be put, no matter whether he had conceived the idea of the use or not."). Here, the receiver/computer must be capable of creating a digest on data that is "in" the receiver's network cache memory.

We find that Santos describes a computer that can calculate a digest on data corresponding to data that will be stored in the receiver's network cache. When data not yet in Santos's compressor's cache is to be sent across the communication channel, Santos's compressor sends that data to the decompressor. Ex. 1004, 7. Santos's decompressor, upon receiving the data also creates a digest for that data and stores the digest and the data in the decompressor's cache. *Id*. Santos describes this case as follows:

> When the compressor receives a packet {HdrA, X} [i.e., data] to be forwarded over the link, where HdrA is the TCP/IP header and X is the data payload, it first computes H(X) [i.e., a digest], the fingerprint of X. If it finds that no entry indexed by H(X) [digest] exists in its cache, it stores X in its cache, indexed by H(X) [digest]. It then forwards the TCP/IP packet across the link. Upon receiving a TCP/IP packet forwarded over the channel, the decompressor also computes H(X) [digest], and stores X in its cache, indexed by H(X) [digest]. The TCP/IP packet is then output from the system.

Ex. 1004, 7, Figure 4.

Santos's decompressor thus calculates a digest for every data payload that it receives from the compressor and stores that data payload and its digest in its network cache memory. Ex. 1004, 7–8. We therefore reject Proxyconn's second argument and find that Santos describes a receiver/computer that includes "means for creating digital digests on data in said network cache memory." We determine that Microsoft has established by a preponderance of evidence that Santos anticipates claims 1 and 3.

(2) Claim 10

Proxyconn contends that Microsoft never challenged the patentability of claim 10 as anticipated by Santos and that our decision to institute a trial on this ground was "a mistake." Resp. 32–33, n.5. For this reason, Proxyconn never substantively addresses Santos as it relates to claim 10. *Id*. at 32–33. However, Microsoft's claim charts compared claim 10 to Santos. '026 Petition, App'x. A 11–13. That claim chart is part of Microsoft's petition, and we instituted review of claim 10 as anticipated by Santos on the grounds raised in the claim chart. Microsoft has thus proffered evidence to establish that Santos describes every limitation of claim 10. Proxyconn fails to rebut Microsoft's evidence. In the absence of countervailing evidence and argument, we determine that Microsoft has established by a preponderance of evidence that Santos anticipates claim 10.

(3) Claim 22

Claim 22 is directed to a method "performed by a receiver/computer." Ex. 1002, col. 12, ll. 30–31. Just as with claims 1 and 3, Proxyconn argues that Santos fails to describe a "receiver/computer." Resp. 33–34. For the same reasons described above in connection with claims 1 and 3, we find that Santos describes the claimed receiver/computer. Microsoft has proffered evidence to establish that Santos meets every other limitation of claim 22, and Proxyconn fails otherwise to rebut that evidence. Therefore, we determine that Microsoft has established by a preponderance of evidence that Santos anticipates claim 22.

(4) Claim 23

Proxyconn argues that Santos fails to describe "searching in predetermined locations of said permanent storage memory for data" for two reasons. Resp. 35–36. First, Proxyconn argues that Santos "lacks permanent storage memory," which is evident from the loss of "fingerprints H(X) and associated data" when Santos's computers are reset. *Id*. at 35 (citing "Ex. 1004 at § 3.2.1").[4] Proxyconn contends "[i]f either the data payload or fingerprints H(x) were stored in permanent storage memory, the data and fingerprint H(x) would be retained in memory, following either a reset or power cycle." Resp. 35–36.

In response, Microsoft characterizes Santos's compressor and decompressor as "general-purpose" computers that "necessarily" have ROM and a hard disk. MS Reply 11. Microsoft also contends that Santos' description of a 200 MB cache in a system with only 128 MB RAM implies that some cache must be in non-volatile memory. *Id*. (citing Ex. 1004, 7–9 and Figures 4 and 5). Microsoft does not directly address Proxyconn's contention that the loss of data in Santos's cache upon a reset demonstrates that Santos's cache does not reside in permanent storage memory.

Based on our own review of Santos, we reject Proxyconn's argument as not being supported by the evidence. Santos's only description of a reset relates to a "reset message" rather than a system reset or power interruption. Ex. 1004, 7 and 9. Santos describes using a reset message sent from the compressor to the decompressor or vice versa as a mechanism for handling a

[4] The cited portion of Santos does not address the effect of a "reset." Nonetheless, Santos describes a reset process elsewhere. Ex. 1004, 9 (in § 3.3).

lack of synchronization between the caches in each machine. *Id*. at 9. We understand these reset messages to be sent intentionally to reset the other cache to a known state in the event that the stored information about "illegal fingerprints" is lost (e.g., due to a compressor restart). *Id.* Santos does not describe where or how the compressor stores information relating to illegal fingerprints. Thus, we find Proxyconn's cited evidence to be inconclusive regarding the character of the cache memory.

Microsoft's evidence on the nature of the cache memory is more persuasive. Santos describes its compressor and decompressor in some detail as being "Intel-based Pentium II 300 MHz machines running Linux 2.0.31 with 128 MB of RAM." Ex. 1004, 9. Santos also states: "we limited the amount of memory available for the caches, . . . , to 200 MB each." *Id*. Santos also expressly describes repeated and numerous write operations to the cache. Ex. 1004, Figures 4 and 5 and accompanying text at 7–8.

We are persuaded that a skilled artisan would understand that Santos's 200 MB cache, which exceeds the available RAM in each machine, referred to a non-volatile memory that supports multiple write operations, which satisfies our interpretation of "permanent storage memory." We therefore reject Proxyconn's argument that Santos does not describe "permanent storage memory."

Second, Proxyconn argues that Santos fails to "search in predetermined locations" within the permanent storage memory. Resp. 36. Proxyconn asserts that "[s]earching in an index or files stored in memory is far different than searching at a predetermined location in memory, much less in predetermined locations in permanent storage memory." *Id*. The sole

support that Proxyconn provides for this assertion is the following testimony of Dr. Konchitsky.

> 51. I understand Santos to disclose that the fingerprints are not stored in permanent storage memory. Santos states that upon reset, such as through a power cycle or restart, the fingerprints H(x) and associated data is lost. [EX1004] at §3.3. This indicates to me, as it would to any person of ordinary skill in the art[,] that the fingerprints are not stored in permanent storage memory, and thus Santos could not logically teach searching permanent storage memory for fingerprints.

Ex. 2002 ¶ 51, Resp. 36. Dr. Konchitsky's testimony does not support Proxyconn's assertion that Santos searches "in predetermined locations" in memory. Rather, Dr. Konchitshy's testimony relates to whether Santos describes "permanent storage memory."

Microsoft responds that Santos "looks in at least two predetermined locations: it looks at H(X) and then, to identify any collisions, it looks at the stored payload associated with the H(X) and compares that to the argument payload." MS Reply 11 (citing Ex. 1004, 7–9, Figures 4 and 5). We determine that the evidence cited by Microsoft is persuasive and, therefore, reject Proxyconn's argument that Santos does not perform "searching in predetermined locations."

For the foregoing reasons, we conclude that Microsoft has established by a preponderance of evidence that Santos anticipates claim 23.

d. DRP

(1) Claims 6, 7, and 9

Proxyconn argues that DRP does not anticipate claims 6, 7, and 9 because it fails to describe the following three limitations on the caching computer recited in claim 6: (i) a "permanent storage memory;" (ii) "means

for comparison;" and (iii) "means for calculating a digital digest" as recited in claim 6. Resp. 37–38. Proxyconn argues that DRP also fails to describe additional limitations on the caching computer that are recited in dependent claims 7 and 9.[5] However, those arguments rely upon Proxyconn's contention that DRP fails to disclose a network cache in permanent storage memory. Resp. 38–39. Microsoft contends that DRP describes the claimed caching computer as any of its client and server computers, all of which have "permanent storage memory," "means for comparison," and "means for calculating a digital digest." MS Reply 8–9. Microsoft also contends that DRP's client and server computers meet each of the additional limitations on the caching computer that are recited in dependent claims 7 and 9.

Proxyconn's argument rests upon Proxyconn's interpretation of "two other computers" as excluding the "caching computer." We have rejected Proxyconn's interpretation as explained in Part II.A.4 above. Proxyconn does not otherwise dispute Microsoft's characterization of DRP's client or server computers as meeting all the limitations on the "caching computer" that are recited in claims 6, 7, and 9. Therefore, we determine that Microsoft has established by a preponderance of evidence that DRP anticipates claims 6, 7, and 9.

[5] Claim 7 depends upon claim 6 and further recites "said caching computer further includes means for calculating a digital digest for data in its network cache memory." Ex. 1002, col. 11, ll. 13–15. Claim 9 also depends upon claim 6 and further recites "said caching computer further includes means for storing said digital digest in said permanent storage memory." *Id*. at col. 11, ll. 18–20.

(2) Claims 11, 12, and 14

Microsoft identifies in great detail how DRP's clients and servers communicate according to the methods of claims 11, 12, and 14. '109 Petition, Appendix A 16–22; MS Reply 9–10 (citing DRP. 4:37–5:19, 5:22–32, 6:40–7:1, 7:20–29, 7:31–35, 7:37–39, 8:11–13, 8:29–31, 9:22–32; Konchitsky Tr. 91:18–94:7, 98:5–11, 108:11–109:18). Proxyconn argues that DRP fails to describe the step of "receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer" as recited in independent claim 11. Resp. 40. Proxyconn contends that DRP's statement that the "client can use the index to automatically download the files that are specified" means that the client downloads specific files without ever sending a response signal to the server. *Id*. Proxyconn also argues that DRP fails to describe the requirement in claim 14 of "a response signal is sent containing a separate indication signal for each of said data objects." *Id*. at 41.

We reject Proxyconn's arguments regarding both claims. DRP describes that the client, after comparing the received digest of files with the digest for its cached versions, sends a GET request (when none of the digests for files match), a differential GET request (when some, but not all, of the digests for files in the cache match), or no request (when all digests for the files in the cache match). DRP, 5:22–32, 6:44, 7:20–28, 8:11–13. Dr. Konchitsky confirms the accuracy of Microsoft's position regarding the GET and differential GET requests that the client might send to the server. Konchitsky Tr. 108:11–109:18. The three types of responses that the client

sends to the server after receiving the index from the server and comparing it to the local index constitute the claimed "response signal containing a positive, partial or negative indication signal" of claim 11. These types of responses also correlate to the "separate indication signal" of claim 14. Proxyconn proffers no argument independently distinguishing claim 12 from DRP. Therefore, we conclude that DRP anticipates claims 11, 12, and 14.

2. Obviousness

"Section 103 [of 35 U.S.C.] forbids issuance of a patent when 'the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.'" *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). To establish obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *See CFMT, Inc. v. Yieldup Int'l Corp.*, 349 F.3d 1333, 1342 (Fed. Cir. 2003); *In re Royka*, 490 F.2d 981, 985 (CCPA 1974).

a. Yohe and Perlman

Microsoft contends that the combination of Yohe and Perlman renders claims 1, 3, 10, and 22–24 obvious under 35 U.S.C. § 103(a). Dr. Long testified that a skilled artisan would have considered Yohe and Perlman to be closely related technologies that are natural to combine because both references address the same problem and use the same algorithm in similar applications. Ex. 1007, 9:12–10:13. Dr. Long also testified that Perlman expressly suggests that its technology is suited for use in any type of node in a network for which synchronization of data is important. *Id*. at 11:1–7.

Proxyconn does not address the sufficiency of the combined teachings of Yohe and Perlman regarding the subject matter of claims 1, 3, 10, and 22–24. Instead, Proxyconn first argues that Perlman is not a proper reference to consider in an obviousness analysis because a skilled artisan would not consider Perlman to be analogous art to the claimed invention. Resp. 28–31. Dr. Konchitsky testified that the '717 Patent addresses increasing data access speed by conserving bandwidth while Perlman uses additional bandwidth to keep its nodes synchronized. Ex. 2002, 6. Dr. Konchitsky concluded that a skilled artisan would not consider Perlman to describe a viable way of increasing data access.

Dr. Long testified that Perlman, Yohe, and the '717 Patent all address the same problem: "the desire to reduce redundancy in network data transmissions where dynamic data previously sent over the network has been stored by the receiver for possible later reuse." Ex. 1007, 9:13–15. We find Dr. Long's statement of the problem addressed by the '717 Patent, Perlman, and Yohe to be persuasive. The '717 Patent states: "The performance gains realized by the present invention are derived from the fact that computers in common wide-area networks tend to repetitively transmit the same data over the network." Ex. 1002, col. 6, ll. 17–20. Both Perlman and Yohe describe reducing the use of bandwidth for data transmission as a way of improving network performance. Ex. 1007, 9:19–10:4 (citing Ex. 1003, col. 3, ll. 52–55; Ex. 1005, col. 4, ll. 32–40). Therefore, we reject Proxyconn's argument that a skilled artisan would not consider Perlman's teachings to be pertinent to the invention.

Proxyconn also argues that it would not have been obvious to incorporate Yohe's permanent storage memory into Perlman's router

because such memory would serve no function in Perlman's router. Resp. 31–32. "The test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference. . . . Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art." *In re Keller*, 642 F.2d 413, 425 (CCPA 1981) (citations omitted). Dr. Long testified at length about the reasons why a skilled artisan would have used permanent storage memory as taught by Yohe for the cache in Perlman. Ex. 1007, 12:4–15:9. Proxyconn proffers no persuasive evidence in support of its position. Therefore, we reject Proxyconn's argument that a skilled artisan would not have found it obvious to use permanent storage memory in the receiver/computer and sender/computer.

For the combination of Yohe and Perlman to render claims 1, 3, 10, and 22–24 obvious, the combination still must describe or suggest all limitations of the claims. Therefore, we analyze the teachings and the differences, if any, between the combination of Yohe and Perlman and the claims below.

(1) Claims 1, 3, and 10

As discussed in Part II.C.1 above, we determine that Yohe anticipates claim 10, but that neither Yohe nor Perlman anticipates claims 1 and 3. The only limitation recited in claims 1 and 3 not described by Yohe is the "means for comparison of digital digests on data" located in the receiver/computer. *See* Part II.C.1.b(1) above. However, Yohe describes comparator 58 in cache verifying computer 14 that would, if located in the remote client, meet the recited "means for comparison of digital digests." Additionally, Perlman describes performing the comparison of digital

digests on data in the receiver because Perlman's receiving router compares a received digest with a locally generated and stored digest.

Ex. 1003, col. 7, ll. 56–63.

Thus, we must decide whether a skilled artisan would have found it obvious to incorporate Perlman's comparing means into Yohe's remote client or to move or add Yohe's comparator 58 from cache verifying computer 14 into remote client 12. Both Perlman and Yohe describe devices that compare digital digests on data. Ex. 1003, col. 7, ll. 56–63; Ex. 1005, col. 6, ll. 22–37, Figures 6 and 7. Yohe suggests that "[i]t is recognized that other locations for the comparator [58] may exist." Ex. 1005, col. 13, ll. 32–34. Proxyconn has not suggested that Yohe's comparator would not be capable of performing the recited function of "comparison of digital digests on data." Incorporating Perlman's comparing means into Yohe's remote client or moving Yohe's comparator 58 into the remote client would have involved nothing more than ordinary skill and would have been using known devices to perform known functions to yield predictable results. We conclude that a skilled artisan would have found it obvious to include the recited "means for comparison of digital digests on data" in a receiver/computer. For these reasons, we conclude that Microsoft has established by a preponderance of evidence that the combination of Yohe and Perlman renders claims 1, 3, and 10 unpatentable as obvious under 35 U.S.C. § 103(a).

(2) Claims 22–24

As explained in Parts II.C.1.a(2) and II.C.1.b(3) above, neither Perlman nor Yohe describes, "searching" as required by claim 22. We cannot conclude that a claim would have been obvious when the prior art

does not describe every element recited in the claim. Claims 23 and 24 depend from claim 22. Therefore, we conclude that Microsoft has not established by a preponderance of evidence that claims 22–24 are unpatentable as obvious under 35 U.S.C. § 103(a).

b. Mattis and DRP

For the reasons expressed in Part II.C.1.d above, we determine that DRP anticipates claims 6, 7, 9, 11, 12, and 14. Because we cancel these claims based on DRP alone, we determine that Microsoft's challenge to the patentability of claims 6, 7, 9, 11, 12, and 14 as being obvious in light of Mattis combined with DRP is moot and do not address this challenge.

D. Proxyconn's Motion to Amend

Proxyconn moved to substitute claims 35–41[6] for challenged claims 1, 3, 6, 10, 11, 22, and 23, respectively, if the Board were to cancel any of those challenged claims as unpatentable. Mot. Amend 1. Proxyconn also requests that we enter claims 35–41 "in addition to the original claims." *Id*. Proxyconn may not add a proposed claim while retaining the original claim for which the proposed claim is substituted. 35 U.S.C. § 316(d)(1)(B) and 37 C.F.R. § 42.121(a)(3). Therefore, to the extent that Proxyconn's Motion to Amend requests entry of substitute claims in addition to the original claims, we deny the Motion.

[6] Proxyconn mistakenly refers to substitute claims 35–42 in its Motion to Amend, but proffers only claims 35–41 as substitutes for claims 1, 3, 6, 10, 11, 22, and 23, respectively. Mot. Amend, Appendix A.

1. Proxyconn's Burden of Persuasion Relating to Claims 35–41

Inter partes review is neither examination nor reexamination of a patent application. We do not enter proposed substitute claims as a matter of right. Rather, the patent owner must prove its entitlement to the proposed claims. Rule 42.20(c) states: "(c) *Burden of proof.* The moving party has the burden of proof to establish that it is entitled to the requested relief." 37 C.F.R. § 42.20(c). We set forth the requirements for demonstrating the prima facie patentability of substitute claims in *Idle Free Sys., Inc. v. Bergstrom, Inc.*, IPR2012-00027, Paper 26, as follows:

> A patent owner should identify specifically the feature or features added to each substitute claim, as compared to the challenged claim it replaces, and come forward with technical facts and reasoning about those feature(s), including construction of new claim terms, sufficient to persuade the Board that the proposed substitute claim is patentable over the prior art of record, and over prior art not of record but known to the patent owner. The burden is not on the petitioner to show unpatentability, but on the patent owner to show patentable distinction over the prior art of record and also prior art known to the patent owner. Some representation should be made about the specific technical disclosure of the closest prior art known to the patent owner, and not just a conclusory remark that no prior art known to the patent owner renders obvious the proposed substitute claims.
>
> A showing of patentable distinction can rely on declaration testimony of a technical expert about the significance and usefulness of the feature(s) added by the proposed substitute claim, from the perspective of one with ordinary skill in the art, and also on the level of ordinary skill, in terms of ordinary creativity and the basic skill set. A mere conclusory statement by counsel, in the motion to amend, to the effect that one or more added features are not described in any

> prior art, and would not have been suggested or rendered obvious by prior art, is on its face inadequate.

Idle Free Sys., Inc. v. Bergstrom, Inc., IPR2012-00027, slip op. at 7–8 (PTAB June 11, 2013 (Paper 26)); *see also Idle Free Sys., Inc. v. Bergstrom, Inc.*, IPR2012-00027, slip op. 33–38 (PTAB January 7, 2014 (Paper 66)).

Proxyconn has not proffered sufficient arguments or evidence to establish a prima facie case for the patentability of claims 35–41. For example, Proxyconn has not: (i) construed the newly added claim terms; (ii) addressed the manner in which the claims are patentable generally over the art; (iii) identified the closest prior art known to it; (iv) addressed the level of ordinary skill in the art at the time of the invention; or (v) discussed how such a skilled artisan would have viewed the newly recited elements in claims 35–41 in light of what was known in the art. Instead, Proxyconn attempts to distinguish claims 35–41 only from the prior art for which we instituted review of corresponding claims 1, 3, 6, 10, 11, 22, and 23. Mot. Amend 4–15. Consequently, Proxyconn has failed to establish a prima facie case for the patentability of claims 35–41. Proxyconn's motion to amend in connection with claims 35–41 is, therefore, denied on these grounds. We also find Proxyconn's motion unavailing for additional reasons raised by Microsoft as described below.

2. Patentability of Claims 35–41 in light of DRP

a. Claims 35, 36, 38, 40, and 41

Microsoft contends that DRP anticipates claims 35, 36, 38, and 40–41 and supports its contentions with detailed citations to DRP. Microsoft Corporation's Opposition to Patent Owner's Corrected Motion to Amend under 37 C.F.R. § 42.121 (Paper 48) 1–15 ("MS Amend Opp."). In its

Motion to Amend, Proxyconn does not compare claims 35, 36, 38, 40, and 41 to DRP. Mot. Amend 4–13. Proxyconn argues that Microsoft improperly injects a "new ground of review" into the trial by asserting that DRP would anticipate claims 35, 36, 38, 40, and 41 because these claims are amended versions of claims 1, 3, 10, 22, and 23 respectively, for which no challenge based on DRP has been instituted. Mot. Amend Reply 2–3.

Proxyconn's argument is unpersuasive. "Petitioners may respond to new issues arising from proposed substitute claims including evidence responsive to the amendment. 35 U.S.C. 316(a) and 326(a). This includes the submission of new expert declarations that are directed to the proposed substitute claims." Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (August 14, 2012).[7] Microsoft has provided evidence—i.e., specific citations to portions of DRP—that responds to new issues introduced by Proxyconn's proposed substitute claims. Microsoft is entitled to do so. Proxyconn provides no evidence to counter Microsoft's contentions that DRP anticipates claims 35, 36, 38, 40, and 41. Proxyconn carries the burden of proof with respect to the patentability of its proposed claims. 37 C.F.R. § 42.20(c). Because the only evidence of record supports Microsoft's position, we conclude that Proxyconn has failed to establish by a preponderance of evidence that claims 35, 36, 38, 40, and 41 are patentable

[7] Proxyconn's Motion to Amend was filed after publication of our decision in *Idle Free Sys., Inc. v. Bergstrom, Inc.*, IPR2012-00027, Paper 26, discussed above, which set forth the requirements for meeting the burden of proof on a motion to amend. We also reminded Proxyconn at the oral hearing of its duty to distinguish the proposed claims from all prior art of which it is aware, including DRP. Tr. 64:3–13.

over DRP. Therefore, we deny the Motion to Amend as it relates to claims 35, 36, 38, 40, and 41.

b. Claim 37

Claim 37, which is substituted for claim 6, is reproduced below with additions indicated by underlining.

> 37. A system for data access in a packet-switched network, comprising:
>
> a gateway including an operating unit, a memory and a processor connected to said packet-switched network in such a way that network packets sent between at least two other computers pass through it;
>
> a caching computer connected to said gateway through a fast local network, wherein said caching computer includes an operating unit, a first memory, a permanent storage memory and a processor; said caching computer further including a network cache memory in its permanent storage memory, means for calculating a digital digest <u>on data</u> and means for comparison between a digital digest on data in its network cache memory and a digital digest received from said packet-switched network through said gateway<u>, wherein said data includes a plurality of octet ranges in a file or files</u>.

Mot. Amend, App. A, 1–2.

We have determined that DRP anticipates claim 6. *See* Part II.C.1.d(1) above. Proxyconn's entire argument for distinguishing claim 37 from DRP is: "Substitute claim 37 requires structure to operate a data including a plurality of octet ranges in a file or files. DRP discloses content identifier based on information objects . . . . Lacking this additional element, DRP . . . fail[s] to anticipate and/or render obvious Substitute Claim 37." Mot. Amend 8–9. Proxyconn fails to cite any support for its characterization of DRP. Microsoft responds: "The data processed in DRP

includes plural files and thus includes a plurality of ranges of octets in plural files. . . . The 'data that is distributed . . . can consist of any kind of code or content.'" MS Amend Opp. 6 (citing DRP, 2:31–32, 2:44–3:2, 3:14–16, 3:28–31) (internal citation omitted). Because we agree with Microsoft, we determine that Proxyconn has failed to establish by a preponderance of evidence that claim 37 is patentable over DRP.

c. Claim 39

Claim 39, which is substituted for claim 11, is reproduced below with additions indicated by underlining and deletions indicated by double square bracketing.

> 39. A method performed by a sender/computer in a packet-switched network for increasing data access, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit data to a receiver/computer, the method comprising the steps of:
>
> creating <u>a digital digest on data;</u> [[and]]
>
> <u>receiving a request for said data from the receiver/computer;</u>
>
> <u>in response to the request for data,</u> transmitting <u>said</u> [[a]] digital digest of said data from said sender/computer to said receiver/computer;
>
> receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and
>
> if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer.

Mot. Amend, App., A 2–3.

Proxyconn contends, without citing any portion of DRP in support,[8] that DRP fails to describe "the step of receiving a request for data, and in response to the request for data, transmitting a digital digest for the data." Mot. Amend 10. In response, Microsoft contends that DRP describes this missing step being performed by the server responding to a GET request from the client for an index file with the current version of the index. MS Amend Opp. 10–11 (citing DRP, 5:22–32, 6:43–7:1, 7:20–31, 7:37–38, 8:8–13, 9:22–32). We determine that on the evidence before us, Proxyconn has failed to establish by a preponderance of evidence that claim 39 is patentable over DRP.

3. Alleged Broadening of Scope in Claims 36, 38, 40, and 41

During *inter partes* review, a patent owner may not amend a challenged claim in a manner that enlarges the scope of that claim. 35 U.S.C. § 316(d)(3); 37 C.F.R. § 42.121(a)(2)(ii). Proxyconn states without further discussion that "the amendments herein do not seek to enlarge the scope of the claims of the '717 Patent." Mot. Amend 1. Microsoft argues that Proxyconn's proposed claims 36, 38, 40, and 41 impermissibly enlarge the scope of challenged claims 3, 10, 22, and 23 respectively. MS Amend Opp. 4, 7. We address each of these claims in turn below.

[8] Proxyconn cites section II.F (sic, II.G) of Proxyconn's Patent Owner Response. Mot. Amend 10. However, this section of the Response cites only ¶¶ 52–61 of Dr. Konchitsky's Declaration. Resp. 36–41. None of those paragraphs cites any portion of DRP to support Dr. Konchitsky's testimony. *See* Ex. 2002, Konchitsky Decl. ¶¶ 52–61.

a. Claim 36

Claim 36, which is substituted for claim 3, is reproduced below with additions indicated by underlining and deletions indicated using strikethrough or double square bracketing.

> 36. A system for data access in a packet-switched network, comprising[[:]] a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor; and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor[[,]]; said sender/computer and said receiver/computer ~~communicating through said network;~~ configured to communicate with one another through said network; said sender/computer ~~further~~ includ~~es~~ing means for creating digital digests on data[[;]] stored in said permanent storage memory, and said receiver/computer ~~further~~ including a network cache memory ~~and~~, means for creating digital digests on data in said network cache memory[[;]], and ~~said receiver/computer including~~ means for ~~comparison~~ comparing between digital digests created by the sender/computer and receiver/computer; wherein said receiver/computer further includes means for storing ~~said created~~ at least one of the digital digests created by the sender/computer in its ~~first or~~ permanent storage memory; wherein the data includes at least a range of octets in a file.

See Mot. Amend, App. A, 1.

Microsoft argues that Proxyconn enlarges claim 36 in two ways. First, the claim is allegedly enlarged by changing the requirement that the sender/computer and receiver/computer are "communicating through said network" to two computers that are "configured to communicate with one another through said network." MS Amend Opp. 4. We agree that claim 36 no longer requires that the sender and receiver computers are communicating through the network, which would necessarily also require

that the computers be "configured" to so communicate. Claim 36 more broadly covers sender and receiver computers that are configured to communicate, but are not necessarily communicating. Doing so impermissibly enlarges the scope of claim 3. Therefore, we deny Proxyconn's motion to amend regarding claim 36, and need not reach Microsoft's second argument.

b. Claim 38

Claim 38, which is substituted for claim 10, is reproduced below with additions indicated by underlining and deletions indicated using strikethrough or double square bracketing.

> 38. A system for data access in a packet-switched network, comprising:
>
> a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through a network;
>
> said sender/computer further including means for creating digital digests on data, and
>
> said receiver/computer further including a network cache memory, means for storing [[a]] at least one of said digital digest received from said network in its permanent storage memory, and said receiver/computer configured to search for a digital digest received from the sender/computer, in response to receiving the digital digest, ~~means for comparison between digital digests;~~ wherein said data includes at least a range of octets in a file.

Mot. Amend, App. A, 2.

Microsoft argues that "said receiver/computer configured to search for a digital digest received from the sender/computer, in response to receiving the digital digest" covers subject matter not covered by the receiver/computer's "means for comparison between digital digests" recited in claim 10. MS Amend Opp. 6–7. Microsoft contends, "[f]or example, the receiver might check for a given digest by comparing hashes (or other identifiers) of digital digests not the digital digests themselves." *Id.* at 7. We agree. Claim 38 covers a receiver/computer that can search for a digital digest in ways other than comparing the digital digests themselves. Therefore, we deny Proxyconn's motion to amend regarding claim 38.

c. Claims 40 and 41

Claim 40, which is substituted for claim 22, is reproduced below with additions indicated by underlining and deletions indicated using strikethrough or double square bracketing.

> 40. A method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of:
>
> sending a request for data;
>
> receiving a message containing a digital digest for the requested data from said network;
>
> searching for each received digital digest ~~data with the same digital digest~~ in said network cache memory,
>
> if ~~data having~~ the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting the negative indication signal [[it]] back through said network; and

> creating a digital digest for data received from the sender/computer and stored in said network cache memory.

Mot. Amend, Appendix A 3.

Claim 41, which is substituted for claim 23, is reproduced below with additions indicated by underlining and deletions indicated using strikethrough.

> 41. The method as claimed in ~~claim 22,~~ claim 40, wherein searching in said network cache memory includes ~~further comprising~~ searching in predetermined locations in said permanent storage memory for data with a digital digest ~~substantially~~ identical to the searched one of the digital digests received from said network.

Id.

Microsoft argues that the third and fourth changes to claim 22 reflected in proposed claim 40 impermissibly enlarge the scope of claim 22. Microsoft contends that "[c]laim 40 newly covers methods that look only for matching digests but not for data having those digests. Claim 22 does not cover such a method." MS Amend Opp. 11. Proxyconn responds that "searching specifically for 'each received digital digest,' . . . is narrower than merely searching generally for data by the digital digest." Mot. Amend Reply 5. Proxyconn's argument mischaracterizes claim 22 as requiring "searching generally for data by the digital digest." Claim 22 recites: "searching for data with the same digital digest" not "by the same digest."

The question presented by the parties' arguments is whether it is possible to search for each received digital digest without searching for data having the same digital digest. Microsoft does not identify an example of how one might search for data with the same digest without using the digest. However, Proxyconn does not provide evidence that it is not possible to

search for each received digest without searching for data having the same digest. Without such evidence, we can only evaluate the scope of the claim based on the plain meaning of the terms. On that basis, we conclude that it would be possible to search for each received digital digest without searching for data having the same digital digest. Therefore, it is possible to practice the method recited in claim 40 without practicing the method recited in claim 22. For this reason, we conclude that claim 40 is impermissibly broader than claim 22. The same flaw exists in claim 41, which depends from claim 40. Therefore, we deny Proxyconn's Motion to Amend as it relates to claims 40 and 41.

III. CONCLUSION

Microsoft has established by a preponderance of evidence that claims 1, 3, 6, 7, 9, 10, 11, 12, 14, 22, and 23 are unpatentable as anticipated and claims 1, 3, and 10 are unpatentable as being directed to obvious subject matter. Microsoft has not established by a preponderance of evidence that claim 24 is unpatentable.

IV. ORDER

It is ORDERED that:

Claims 1, 3, 6, 7, 9–12, 14, 22, and 23 are CANCELED; and

Proxyconn's Motion to Amend Claims is DENIED.

Cases IPR2012-00026 and IPR2013-00109
Patent 6,757,717

PATENT OWNER:

Matthew L. Cutler
Bryan K. Wheelock
Douglas A. Robinson
HARNESS, DICKEY & PIERCE, PLC
mcutler@hdp.com
bwheelock@hdp.com
drobinson@hdp.com

PETITIONER:

John D. Vandenberg
Stephen J. Joncus
KLARQUIST SPARKMAN LLP
john.vandenberg@klarquist.com
stephen.joncus@klarquist.com

US006757717B1

(12) **United States Patent**
Goldstein

(10) **Patent No.: US 6,757,717 B1**
(45) **Date of Patent: Jun. 29, 2004**

(54) **SYSTEM AND METHOD FOR DATA ACCESS**

(75) Inventor: **Leonid Goldstein**, Herzlia (IL)

(73) Assignee: **ProxyConn, Inc.**, Irvine, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/398,007**

(22) Filed: **Sep. 16, 1999**

(51) **Int. Cl.**[7] **G06F 15/16**
(52) **U.S. Cl.** **709/217**; 707/1; 707/10; 709/203; 709/225; 709/229
(58) **Field of Search** 709/217, 203, 709/229, 225; 707/10, 1

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,897,781	A		1/1990	Chang et al.
5,384,565	A		1/1995	Gregory
5,682,514	A	*	10/1997	Yohe et al. 707/1
5,732,265	A		3/1998	Dewitt et al.
5,835,943	A		11/1998	Yohe et al.
5,852,717	A		12/1998	Bhide et al.
5,864,837	A		1/1999	Maimone
5,878,213	A	*	3/1999	Bittinger et al. 707/10
5,909,569	A	*	6/1999	Housel et al. 707/10
5,919,247	A	*	7/1999	Van Hoff et al. 709/217
5,924,116	A		7/1999	Aggarwal et al.
5,931,947	A	*	8/1999	Burns et al. 713/201
5,978,791	A		11/1999	Farber et al.
6,003,087	A	*	12/1999	Housel et al. 707/203
6,006,034	A		12/1999	Heath et al.
6,012,085	A		1/2000	Yohe et al.
6,021,491	A	*	2/2000	Renaud 713/179
6,073,173	A	*	6/2000	Bittinger et al. 707/513
6,085,249	A	*	7/2000	Wang et al. 709/229
6,101,543	A	*	8/2000	Alden et al. 709/229
6,122,637	A	*	9/2000	Yohe et al. 709/204
6,134,583	A	*	10/2000	Herriot 709/217
6,148,340	A	*	11/2000	Bittinger et al. 709/203
6,256,664	B1	*	7/2001	Donoho et al. 709/204
6,279,041	B1	*	8/2001	Baber et al. 709/232

OTHER PUBLICATIONS

RFC2068: Hypertext Transfer Protocol—HTTP/1.1, Jan. 1997, pp. 9–12, 29–35, 70–94, 101–138.
RFC791: Internet Protocol Darpa Internet Program Protocol Specification, Sep. 1981, pp. 3, 11–14, 26.
RFC761: DOD Standard Transmission Control Protocol, Jan. 1980, pp.4, 15–17.
RFC1826: IP Authentication Header, Aug. 1995, pp. 1–10.
RFC1864: The Content–MD5 Header Field, Oct. 1995, pp.1–2.
RFC1321: The MD5 Message–Digest Algorithm, Apr. 1992, pp. 1–21.
Dave Fogle, Ethernet Frame Types, article from Feb. 1996 issue of LAN Magazine/Network Magazine, pp. 1–4.
W. Richard Stevens, TCP/IP Illustrated, vol. 1, 1994, pp. 33–37, 143–147, 223–228.

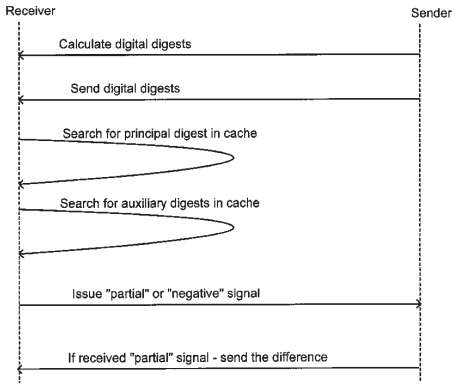
* cited by examiner

Primary Examiner—Jack B. Harvey
Assistant Examiner—Hai V. Nguyen
(74) *Attorney, Agent, or Firm*—J. D. Harriman, II, Esq.; Coudert Brothers LLP

(57) **ABSTRACT**

The invention provides a system for data access in a packet-switched network, including a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, the sender/computer and the receiver/computer communicating through the network; the sender/computer further including device for calculating digital digests on data; the receiver/computer further including a network cache memory and device for calculating digital digests on data in the network cache memory; and the receiver/computer and/or the sender/computer including device for comparison between digital digests. The invention also provides a method and apparatus for increased data access in a packet-switched network.

34 Claims, 9 Drawing Sheets



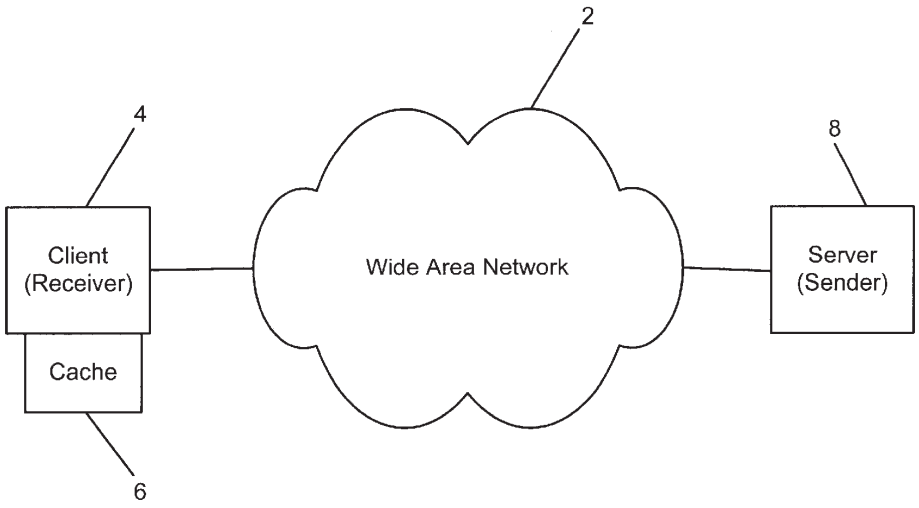


FIG. 1 (PRIOR ART)

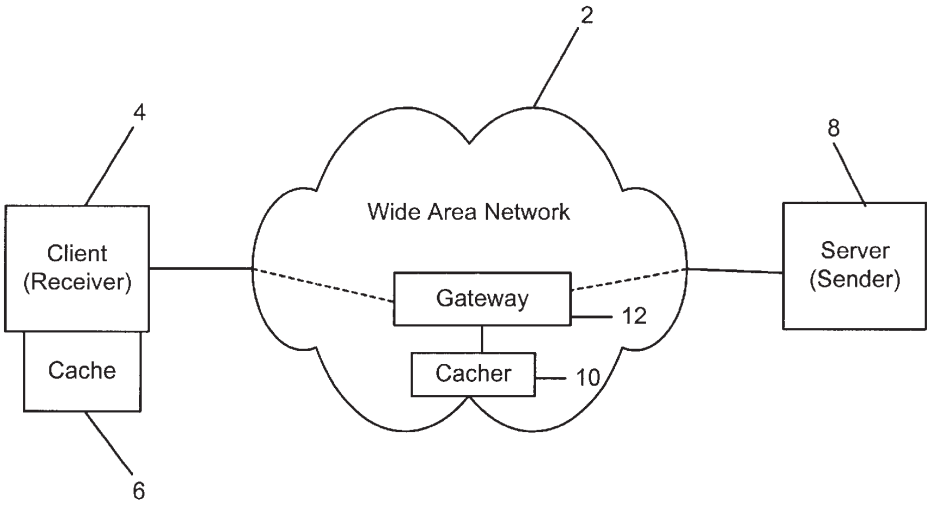


FIG. 2 (PRIOR ART)

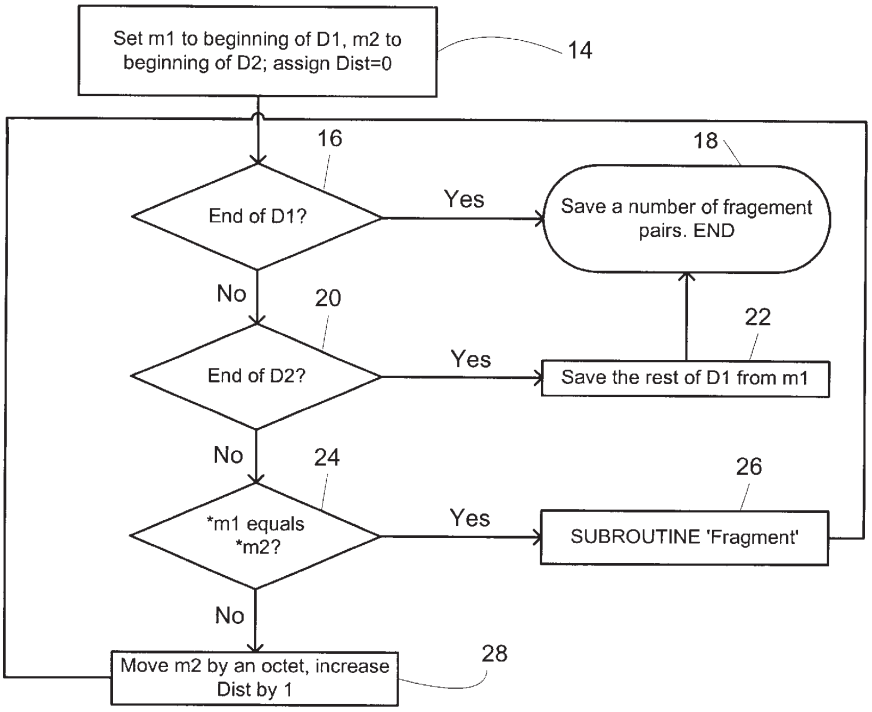
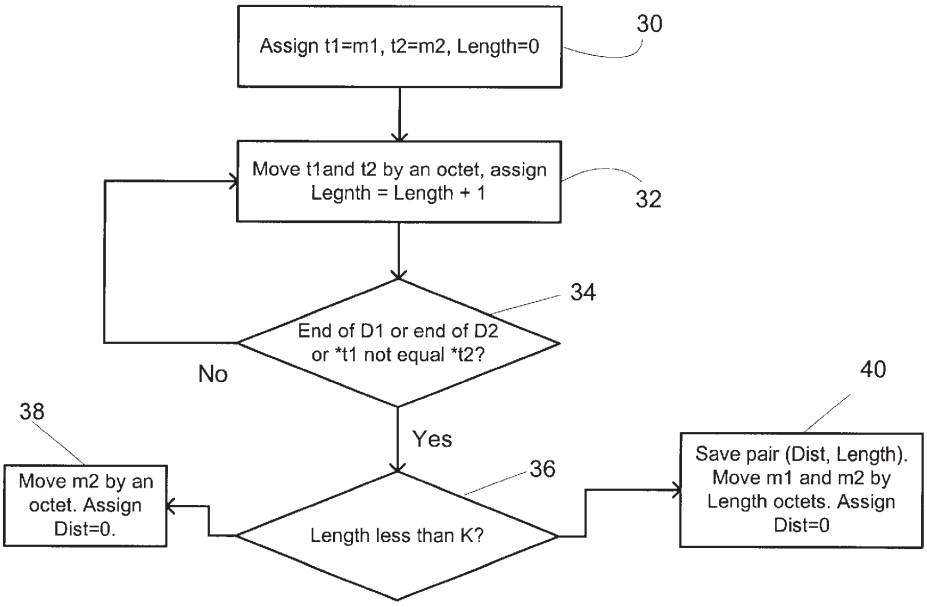


FIG. 3

SUBROUTINE 'Fragment'



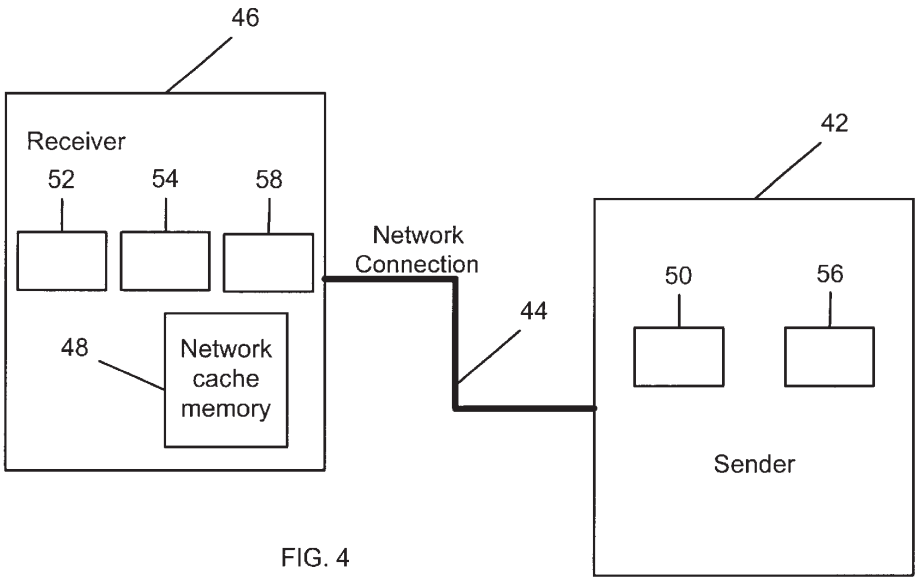

46
Receiver
52
54
58
48
Network cache memory
Network Connection
44
42
50
56
Sender

FIG. 4

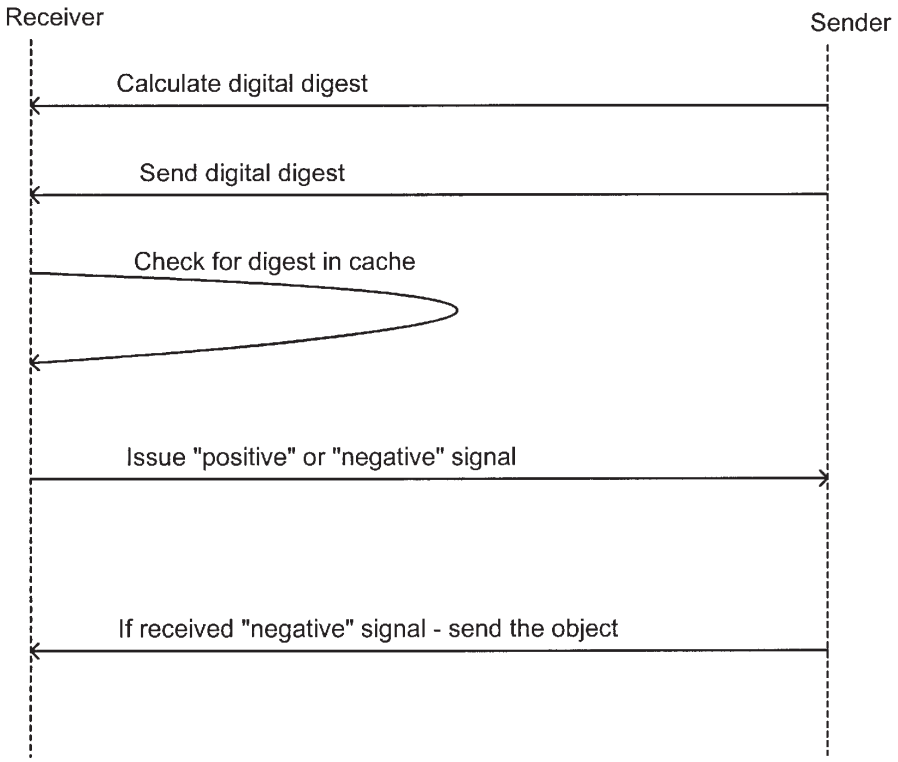

Receiver
Sender
Calculate digital digest
Send digital digest
Check for digest in cache
Issue "positive" or "negative" signal
If received "negative" signal - send the object

FIG. 5

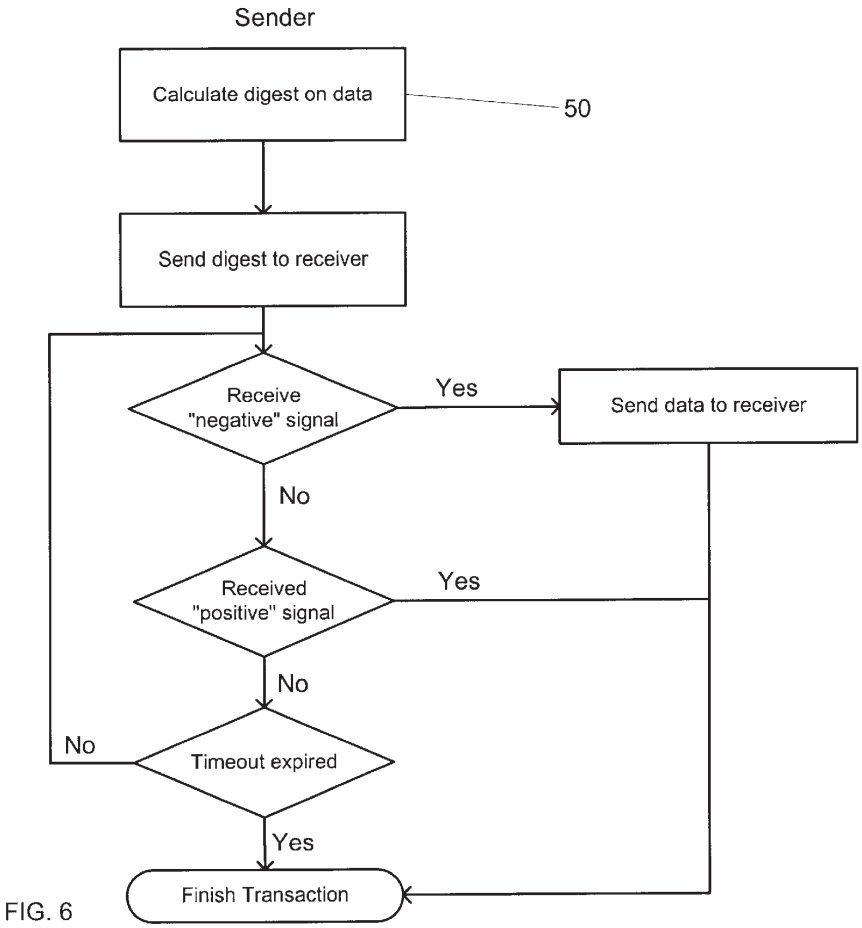

Sender
Calculate digest on data
50
Send digest to receiver
Receive "negative" signal
Yes
Send data to receiver
No
Received "positive" signal
Yes
No
No
Timeout expired
Yes
Finish Transaction

FIG. 6

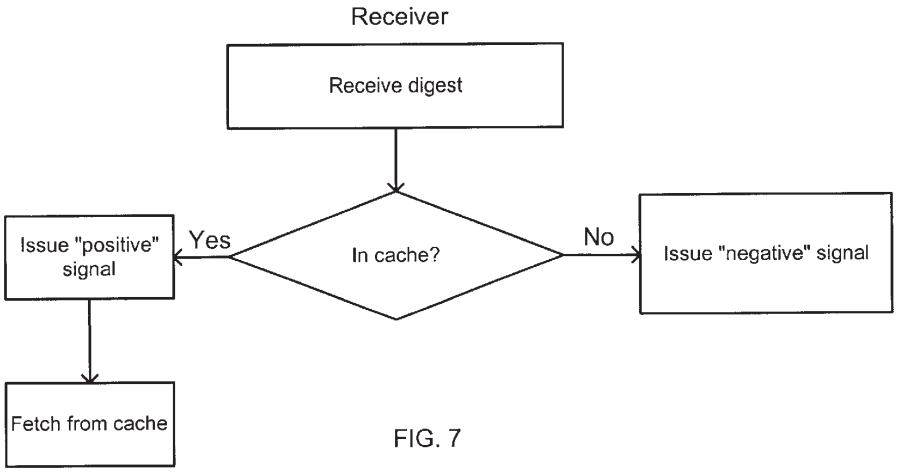

Receiver
Receive digest
Issue "positive" signal
Yes
In cache?
No
Issue "negative" signal
Fetch from cache

FIG. 7

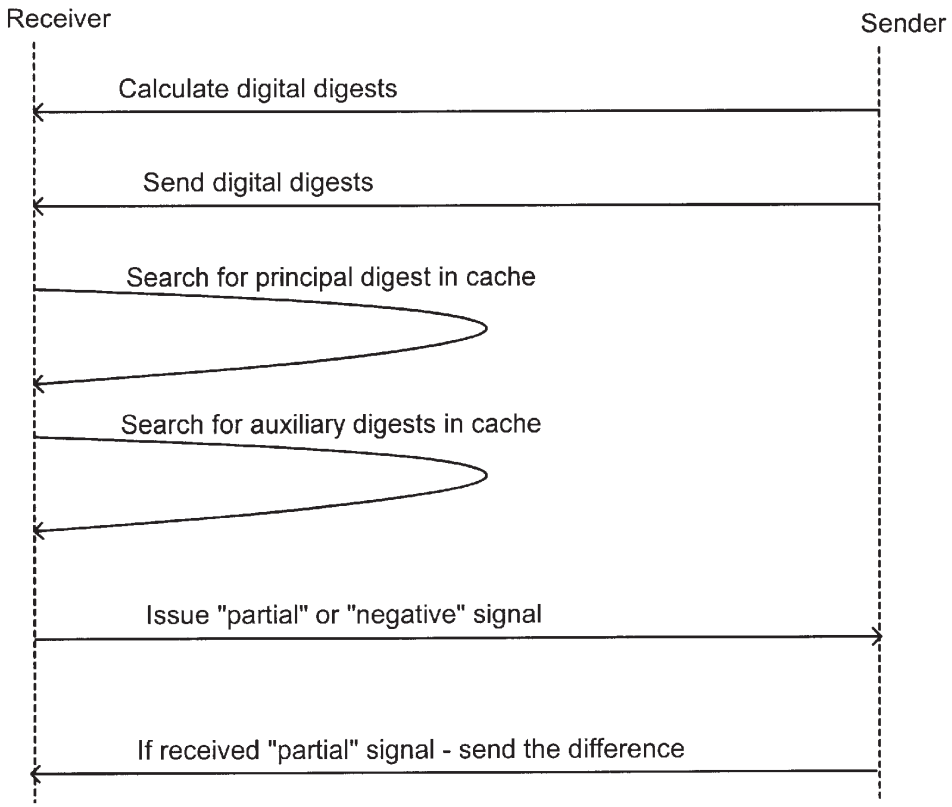

Receiver
Sender
Calculate digital digests
Send digital digests
Search for principal digest in cache
Search for auxiliary digests in cache
Issue "partial" or "negative" signal
If received "partial" signal - send the difference

FIG. 8

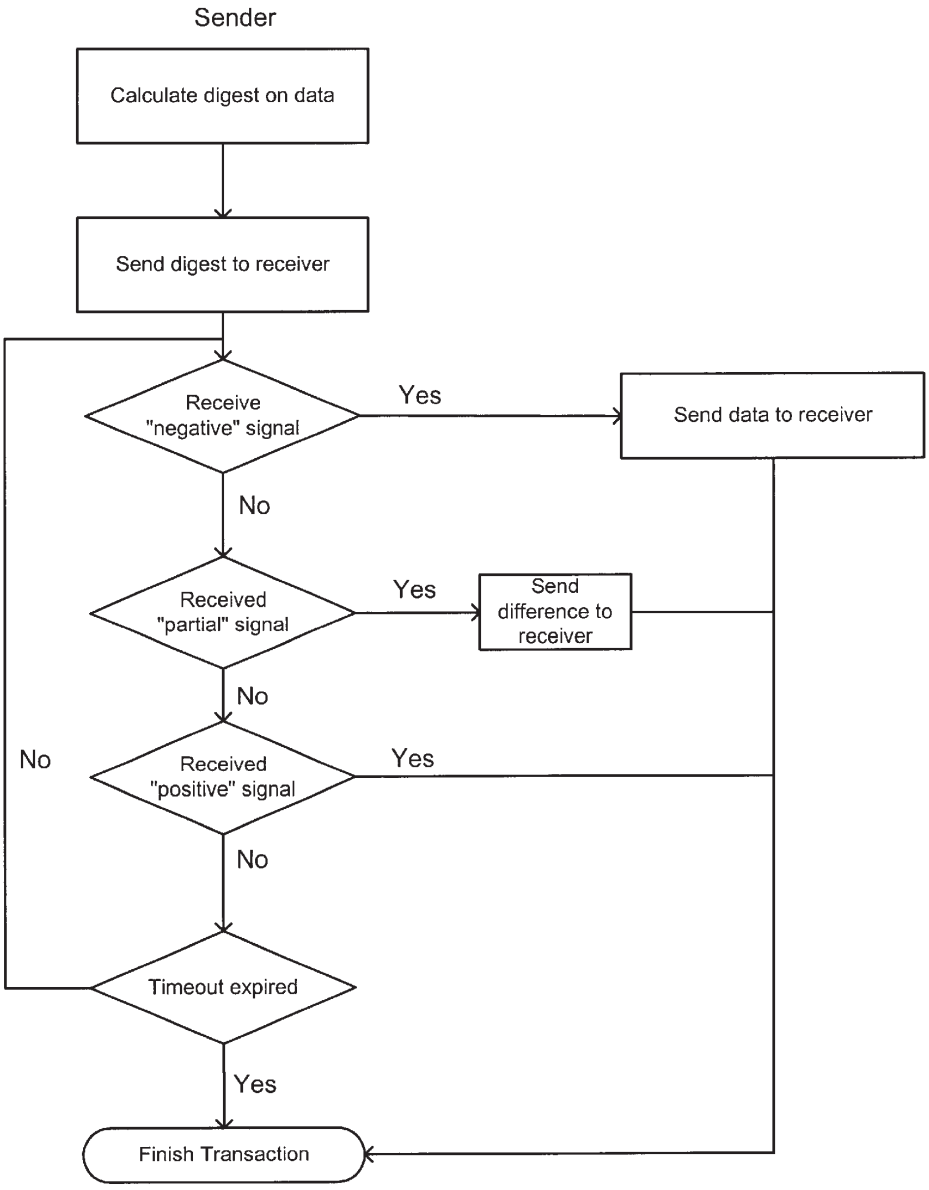

Sender
Calculate digest on data
Send digest to receiver
Receive "negative" signal
Yes
Send data to receiver
No
Received "partial" signal
Yes
Send difference to receiver
No
Received "positive" signal
Yes
No
No
Timeout expired
Yes
Finish Transaction

FIG. 9

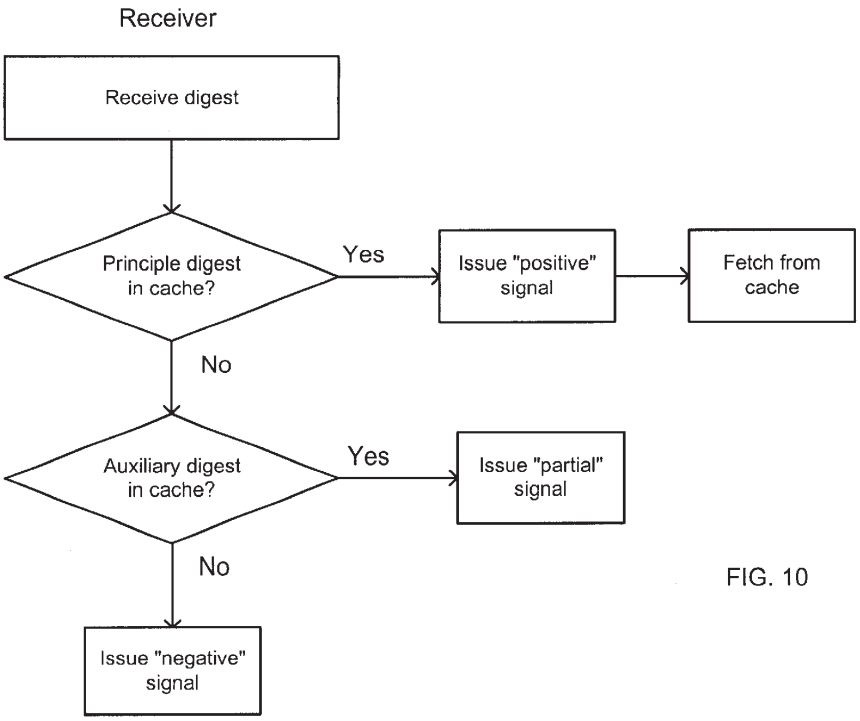

Receiver
Receive digest
Principle digest in cache?
Yes
Issue "positive" signal
Fetch from cache
No
Auxiliary digest in cache?
Yes
Issue "partial" signal
No
Issue "negative" signal

FIG. 10

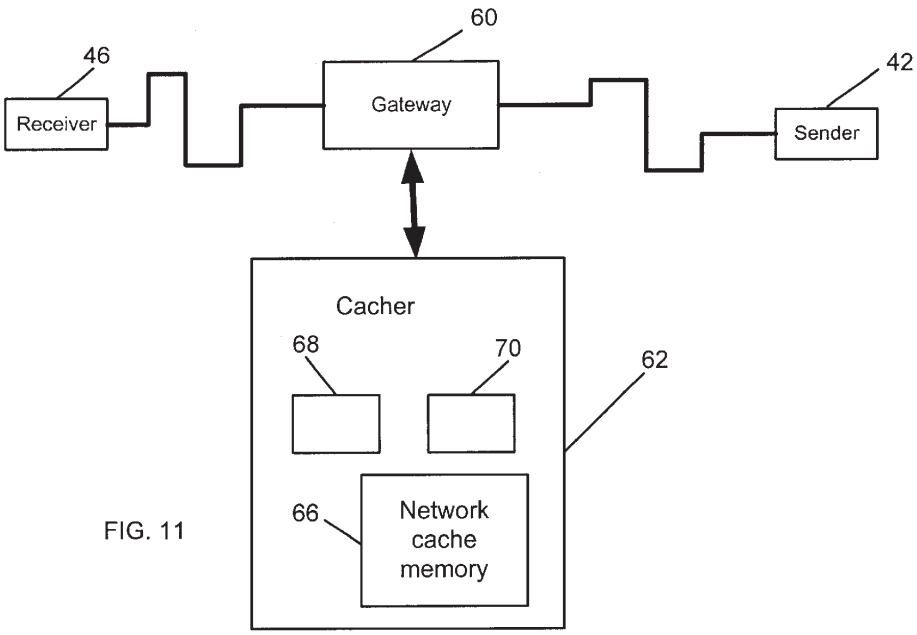

60
46
42
Receiver
Gateway
Sender
Cacher
68
70
62
66
Network cache memory

FIG. 11

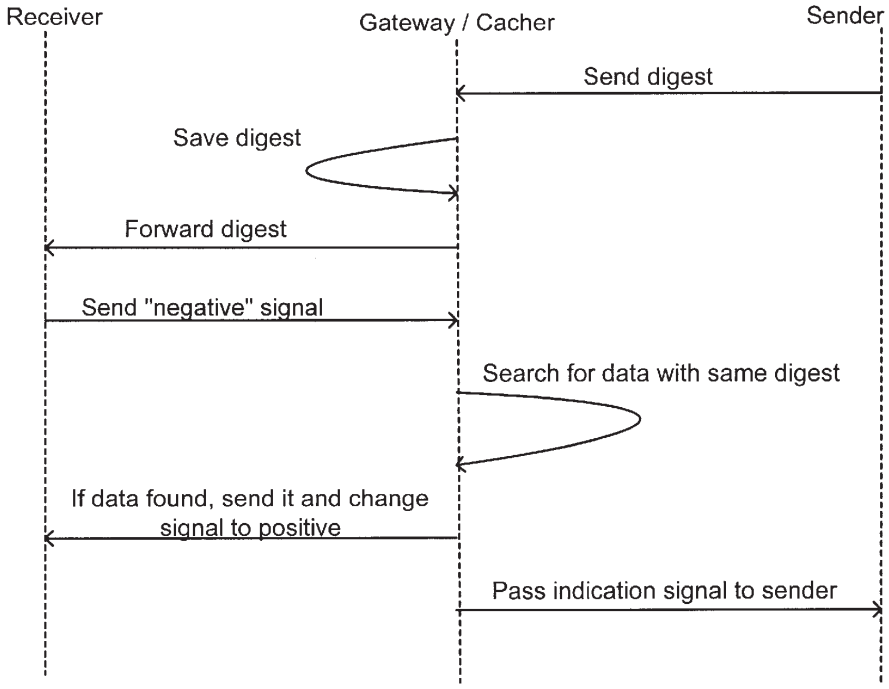

Receiver
Gateway / Cacher
Sender
Send digest
Save digest
Forward digest
Send "negative" signal
Search for data with same digest
If data found, send it and change signal to positive
Pass indication signal to sender

FIG. 12

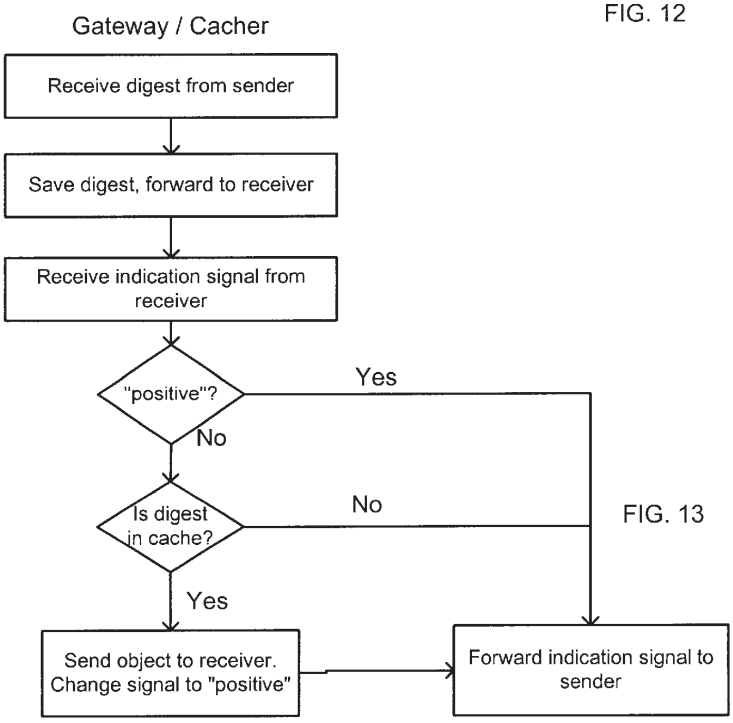

Gateway / Cacher
Receive digest from sender
Save digest, forward to receiver
Receive indication signal from receiver
"positive"?
Yes
No
Is digest in cache?
No
Yes
Send object to receiver. Change signal to "positive"
Forward indication signal to sender

FIG. 13

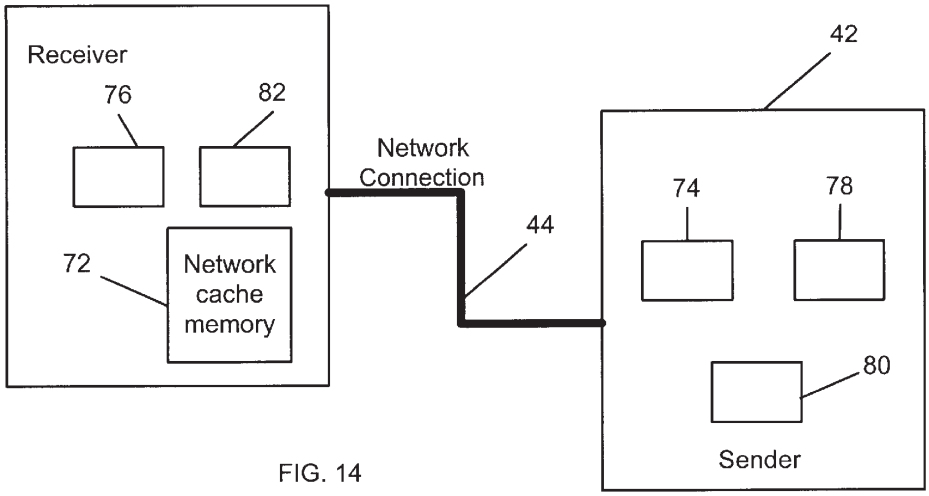

Receiver
76
82
72
Network cache memory
Network Connection
44
42
74
78
80
Sender

FIG. 14

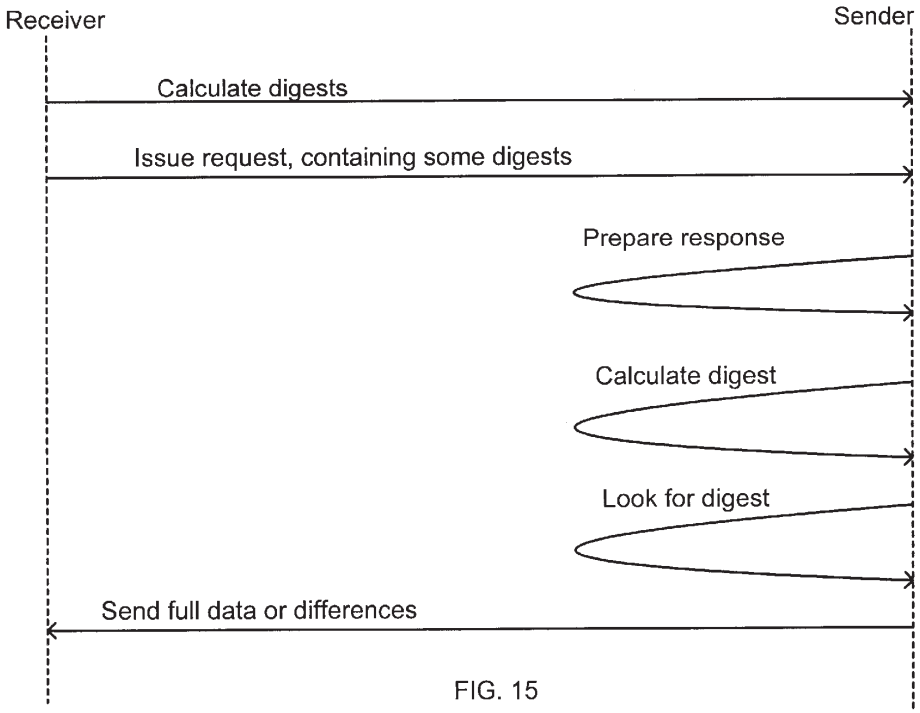

Receiver
Sender
Calculate digests
Issue request, containing some digests
Prepare response
Calculate digest
Look for digest
Send full data or differences

FIG. 15

SYSTEM AND METHOD FOR DATA ACCESS

RELATED APPLICATION

This application claims priority and is entitled to the filing date of Israeli application Ser. No. 126292 filed Sep. 18, 1998, and entitled "System And Method For Data Access," and which describes the same invention as defined herein.

FIELD OF THE INVENTION

The present invention relates to data access in networks. Specifically, the invention is concerned with a method, system and apparatus for increasing the speed of data accessing in communication networks.

BACKGROUND OF THE INVENTION

Many known applications and protocols provide means for caching and verifying of data transmitted via a network **2** (FIG. **1**, prior art). Thus, a client (receiver) **4** caches data received from network **2** in cache **6**. Then, when data from a remote server (sender) **8** is requested, it first searches its local cache. If the requested data is available in the cache and is verified to be valid, the client uses it, and transmission over the network is not required. Gateway or proxy caches **10** (FIG. **2**, prior art) are able to operate in a similar manner.

The most well-known techniques are as follows:

1) In response to a request from a receiver, a sender attaches to the sent data an expiration time in absolute or relative form. The receiver, and possibly proxies, cache the data together with its request until the expiration time. Then the data is retrieved from the cache. In some cases, the receiver guesses the expiration time.

The problem associated with this technique is that the data entity can be changed before the expiration time, and the receiver would use an obsolete version of the data without even knowing it. Also, when the time has expired, the data will be resent, even if it is up to date.

2) In response to a request from a receiver, the sender attaches a validator to the sent data. The validator changes at least every time the data changes; in many cases, system time is used as the validator. The receiver, and possibly proxies, cache the data together with its request. When making the next request for the same data to the same sender, the receiver includes the validator. The sender keeps track of the data and resends it only if it were changed.

The problems associated with this technique are:

a) Data is cached according to requests and senders. If the same request is directed to different servers, cached data cannot be reused.

b) Requests without concrete data cannot be cached.

c) The sender must track the cached data, which is not always possible.

None of the prior art techniques discussed above provides means for transmitting minor differences in data. Additionally, if data is retrieved through a caching proxy, there is a danger that an unauthorized user will have access to the data.

It is therefore a broad object of the present invention to provide a method, system and apparatus for increasing the speed of data access in a packet-switched network.

Another object of the present invention is to decrease data traffic throughout the network.

Still another object of the present invention is to decrease the required cache size.

A yet further object of the present invention is to maintain accessed data integrity and to improve security.

SUMMARY OF THE INVENTION

The terms "data" or "data object" as used herein refer to a file or range of octets in a file, a range of frames in a video stream or RAM-based range of octets, a transport level network packet, or the like.

The term "digital digest" as used herein refers to a fixed-size binary value calculated from arbitrary-size binary data in such a way that it depends only on the contents of the data and the low probability that two different data or objects have the same digital digest.

The term "gateway" as used herein also includes network proxies and routers.

If a sender/computer in a network is required to send data to another receiver/computer, and the receiver/computer has data with the same digital digest as that of the data to be sent, it can be assumed with sufficient probability for most practical applications that the receiver/computer has data which is exactly the same as the data to be sent. Then, the receiver/computer can use the data immediately without its actual transfer through the network. In the present invention, this idea is used in a variety of ways.

In one embodiment of the invention, a sender/computer required to send data to a receiver/computer computer initially sends a digital digest of the data. If the receiver/computer already has data with the same digital digest, it uses this data as if it were actually transmitted from the sender/computer. Additionally, digital digests for other data objects can be sent together with the principal digest. If the receiver/computer cannot find data having the principal digest, it searches for data with one of these auxiliary digests. If such data is found, the sender/computer is required to send only the difference between the requested data object and the data object corresponding to the digest.

The expression "difference between a first data or data object and a second data or data object" as used herein means any bit sequence that enables the restoration of the first data, given the second data, the bit sequence and the method employed in calculating the difference.

The invention may be implemented in a gateway system. Such a system comprises a gateway computer connected to a packet-switched network in such a way that network packets sent between at least two other computers pass through it; a caching computer connected to the gateway computer, the caching computer having a network cache memory in its permanent storage memory, means for calculating a digital digest on the data it stores and means for comparison between a digital digest calculated on data in its network cache memory and a digital digest received from the packet-switched network by the gateway computer. When this system intercepts an indication signal other than a positive indication signal for a certain digital digest from a receiver/computer computer, if it has data with the same digest, it sends this data to the receiver/computer.

In another embodiment of the present invention, a client computer sends to a server computer a request including digital digests. A sender/computer forming a response then searches for data with the same digital digests as those received. If the digest of the data in the response equals one of the received digests, the server only sends confirmation. If the digest of another data is identical to one of the received digests, only the difference(s) between these data is sent.

In accordance with the present invention, there is therefore provided a system for data access in a packet-switched

network, comprising a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through said network; said sender/computer further including means for calculating digital digests on data; said receiver/computer further including a network cache memory and means for calculating digital digests on data in said network cache memory; and said receiver/computer and/or said sender/computer including means for comparison between digital digests.

The invention also provides a system for data access in a packet-switched network, comprising a gateway computer including an operating unit, a memory and a processor connected to said packet-switched network in such a way that network packets sent between at least two other computers pass through it; a caching computer including an operating unit, a first memory, a permanent storage memory and a processor connected to said gateway computer through a fast local network; said caching computer further including a network cache memory in its permanent storage memory, means for calculating a digital digest on data stored therein and means for comparison between a digital digest calculated on data in its network cache memory and a digital digest received from said packet-switched network through said gateway computer.

In addition, the invention provides a system for data access in a packet-switched network, comprising a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through a network; said sender/computer further including means for calculating digital digests on data, and said receiver/computer further including a network cache memory, means for storing a digital digest received from said network in its permanent storage memory and means for comparison between digital digests.

The invention further provides a method performed by a sender/computer in a packet-switched network for increasing data access, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit data to a receiver/computer, the method comprising the steps of transmitting a digital digest of said data from said sender/computer to said receiver/computer; receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer.

The invention still further provides a method for increasing data access performed by a sender/computer in a packet-switched network, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit principal data to a receiver/computer, said method comprising the steps of transmitting digital digests of said principal data and of one or more auxiliary data from said sender/computer to said receiver/computer; receiving a response signal at said sender/computer from said receiver/computer, said response signal containing a positive, negative or partial indication signal, and if a partial indication signal is received, said sender/computer transmitting a signal constituting the difference between said principal data and corresponding auxiliary data.

The invention yet further provides a method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of receiving a message containing a digital digest from said network; searching for data with the same digital digest in said network cache memory, and if data having the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting it back through said network.

Still further, the invention provides a method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of receiving a message containing a digital digest from said network; searching for data with the same digital digest in said network cache memory, and if data having the same digital digest as the digital digest received is uncovered, forming a positive indication signal and transmitting it back through said network.

In addition, the invention provides a method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of receiving a message containing a principal digital digest and one or more auxiliary digital digests from said network; searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to said principal digital digest; searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to one of said auxiliary digital digests; and if data having the same digital digest as one of said auxiliary digital digests received is uncovered, forming a partial indication signal and transmitting it back through said network.

Yet further, the invention provides a method for increased data access performed by a computer system in a packet-switched network, said computer system including a network cache memory and being operationally interposed between a sender/computer and a receiver/computer so that data packets sent between said sender/computer and said receiver/computer are delivered through said computer system; said method comprising the steps of intercepting a message containing a digital digest transmitted from said sender/computer to said receiver/computer, and transmitting data with a digital digest substantially identical to the digital digest received from said sender/computer to said receiver/computer.

In addition, the invention provides a method for increased data access performed by a computer system in a packet-switched network, said computer system including a network cache memory and being operationally interposed between a sender/computer and a receiver/computer so that data packets sent between said sender/computer and said receiver/computer are delivered through said computer system; said method comprising the steps of intercepting a message containing a digital digest transmitted from said sender/computer to said receiver/computer; intercepting a message containing an indication signal other than a positive indication signal transmitted from said receiver/computer to said sender/computer in response to said message containing a digital digest, and transmitting data with a digital digest substantially identical to the digital digest received from said sender/computer to said receiver/computer.

Additionally, the invention provides a method for increased data access performed by a client computer in a packet-switched network, said client computer including an operating unit, a first memory and a processor, said method comprising the steps of sending a request for data from said client computer to a server, said request containing digital digests for different data; said server preparing a response to said request, searching for data with a digital digest substantially identical to one of the digital digests received in said request, and producing the difference between said response and the uncovered data.

Finally, the invention provides apparatus for increased data access in a packet-switched network, comprising a computer connected to said packet-switched network, including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory; means for calculating digital digests of data in said network cache memory; means for comparison between digital digests, and means for sending the results of comparison between a digital digest received from another computer in said network and a digital digest calculated on data in said network cache memory back to said other computer.

The invention will now be described in connection with certain preferred embodiments with reference to the following illustrative figures so that it may be more fully understood.

With specific reference now to the figures in detail, it is stressed that the particulars shown are by way of example and for purposes of illustrative discussion of the preferred embodiments of the present invention only, and are presented in the cause of providing what is believed to be the most useful and readily understood description of the principles and conceptual aspects of the invention. In this regard, no attempt is made to show structural details of the invention in more detail than is necessary for a fundamental understanding of the invention, the description taken with the drawings making apparent to those skilled in the art how the several forms of the invention may be embodied in practice.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** illustrates a prior art wide-area network;

FIG. **2** illustrates a prior art wide-area network with a caching gateway;

FIG. **3** is a flow diagram of the method of calculating the difference between two data digests according to the present invention;

FIG. **4** is a block diagram of a first embodiment of a sender/computer-receiver/computer system according to the present invention;

FIG. **5** is a schematic representation illustrating the interaction between a sender/computer and a receiver/computer according to the system of FIG. **4**;

FIG. **6** is a flow diagram illustrating the method of operating the sender/computer according to the present invention;

FIG. **7** is a flow diagram illustrating the method of operating the receiver/computer according to the present invention;

FIG. **8** is a schematic representation illustrating the interaction between a sender/computer and a receiver/computer according to another embodiment of the present invention;

FIG. **9** is a flow diagram illustrating the method of operating the sender/computer according to a further embodiment of the present invention;

FIG. **10** is a flow diagram illustrating the method of operating the receiver/computer according to the embodiment of FIG. **9**;

FIG. **11** is a block diagram of the configuration of the gateway system according to the present invention;

FIG. **12** is a schematic representation of the interaction between a sender/computer, a receiver/computer, and the gateway configuration according to the present invention;

FIG. **13** is a flow diagram of the operation of the gateway;

FIG. **14** is a block diagram of a further configuration of a sender/computer-receiver/computer system according to the present invention; and

FIG. **15** is a schematic representation of the interaction between the sender/computer-receiver/computer system of FIG. **14**.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

The performance gains realized by the present invention are derived from the fact that computers in common wide-area networks tend to repetitively transmit the same data over the network.

The operations described herein may take the form of electrical or optical signals. The packet-switched network may be Internet.

The term "digital digest" as used herein refers to the per se known MD5 algorithm, described in RFC 1321 by R. Rivest, which is a preferred calculation method. Other algorithms may, however, just as well be used. For example, a digital digest may be calculated according to the CRC algorithm, or by applying the CRC algorithm to different subsets or different reorderings of data, or by consecutively applying CRC and MD5. In addition, any other algorithm may be used, provided that it produces a fixed-size binary value calculated from arbitrarily-sized binary data in such a way that it depends only on the contents of said data and that the probability of two different data having the same digital digest, is low.

Whenever means for calculating the difference between two data are mentioned herein, the method as shown in FIG. **3** may be employed. The data are referred to as D**1** and D**2**. The difference between them consists of three parts: the number of fragment pairs, the array of fragment pairs, and the remainder of D**1**. A fragment pair is a pair representing the distance from the beginning of this fragment to the end of the previous one, and the number of octets in the fragment (Dist,Length). A marker m**1** is set at the beginning of the data D**1** and a marker m**2** at the beginning of D**2**.

An octet m**1** is designated as *m**1** and an octet m**2** as *m**2**. An integer K>1, which represents a minimal length of a fragment encoded, e.g., K=3, is chosen.

As stated above, m**1** is set at the beginning of D**1**, m**2** at the beginning of D**2**, and Dist=0 is assigned at 14. A loop is then entered: if m**1** is at the end of D**1** **(16)**, a number of fragment pairs is saved at 18, and the algorithm is completed. If m**2** is at the end of D**2** **(20)**, the rest of D**1** from m**1** is saved at 22, a number of fragment pairs is saved, and the algorithm is completed. If *m**1** equals *m**2** **(24)**, a subroutine "Fragment" is entered at 26; otherwise, m**2** is moved by one octet toward the end of D**2** and Dist is increased by 1 at 28.

The subroutine "Fragment" proceeds as follows: New markers t1=m1 and t**2**=m**2** are set and Length=0 is assigned at 30. t**1** and t**2** are moved by an octet toward the ends of D**1** and D**2** and Length is increased at 32. If t**1** is at the end of D**1**, or t**2** is at the end of D**2**, or *t**1** does not equal *t**2** at the end of the fragment **(34)**, then the Length is a length of the fragment and Dist is the distance between the beginning of

this fragment and the end of the previous one. If the Length <K as determined by 36, the fragment is dropped at 38, m2 is moved by one octet and the subroutine is terminated. Otherwise, the pair (Dist,Length) is saved, the number of pairs is increased by one, m1 and m2 are moved by Length octets toward the ends of D1 and D2, and Dist is reset to 0 at 40. The subroutine is ended.

The sequence of fragment pairs may be further reduced in size by using the per se known Huffman encoding or by using an arithmetic coding, e.g., as disclosed in U.S. Pat. No. 4,122,440.

Restoration of the data is simple. Marker m2 is set at the beginning of the known D2. Then for each fragment pair (Dist, Length) from the known difference, m2 is moved by Dist octets, Length octets are copied from m2 to D1 and m2 is moved by Length. Then the rest of D1 is copied from the remainder

An embodiment of a sender/computer-receiver/computer system according to the present invention is schematically illustrated in FIG. **4**. A preferred embodiment is a network computer system having at least two computers. A sender/computer **42** (also referred to herein as "sender/computer") having an operating unit, a first memory, a permanent storage memory and a processor, is connected to the network by any network connection **44**. A remote receiver/computer **46** (also referred to herein as "receiver/computer") having an operating unit, a first memory, a permanent storage memory and a processor, is also connected to the network. The receiver/computer **46** uses a part of its permanent storage memory or its first memory, or both, as network cache memory **48**. The sender/computer has calculation means **50** for calculating a digital digest on data in its first memory or in its permanent storage memory. Similarly, the receiver/computer has calculating means **52** for calculating a digital digest on data stored in its network cache memory **48**. The receiver/computer also has comparison means **54** for comparing between such a calculated digital digest and a digital digest received from the network.

An example of a first memory could be a RAM; an example of a permanent storage memory may be a disk drive, a flash RAM or a bubble memory.

It is possible to modify this system in different ways. The receiver/computer **46** and sender/computer **42** may each include means for storing the calculated digital digest in its first memory or permanent storage memory. Additionally, the receiver/computer **46** may have means for calculating a digital digest on data in its permanent storage memory outside of its cache memory. Furthermore, the system may be modified in such a way that the sender/computer **42** has means **56** for calculating the difference between two data objects.

Interaction between the receiver/computer and the sender/computer is depicted in FIGS. **5** to **7**. The data sender/computer **42** calculates a digital digest on the data in means **50** and then transmits the calculated digest to receiver/computer **46**. The receiver/computer receives the digital digest from sender/computer **42** and then searches its network cache memory **48** for data with the same digest. If it finds such data, it uses it as if it were received from the sender/computer **42** and issues a positive indication signal to the sender/computer. Otherwise, it sends a negative indication signal to the sender/computer. Upon receiving a negative indication signal, the sender/computer transmits the data. Upon receiving a positive indication signal, or upon expiration of a predefined period of time, the sender/computer completes the transaction. This transaction begins with a receiver/computer sending a request to the sender/computer.

The above-described method may be modified in different ways. For example, absence of a signal from the receiver/computer for a predetermined period of time may be considered by the sender/computer to be a negative indication signal. Alternatively, the digital digests for some data may be stored in the permanent storage memory of the sender/computer and obtained from there, or a plurality of data may be processed in one transaction, a digital digest being calculated for each data object and a separate indication signal issued on each digital digest.

Another method of interaction between the receiver/computer **46** and the sender/computer **42** is illustrated in FIGS. **8**–**10**. The data sender/computer calculates a digital digest on the data to be sent (hereinafter, "principal digest") and for one or more other data objects (hereinafter, "auxiliary digests"). Without limiting the scope of the invention, the following data objects may be recommended: (a) a previous version of the data requested; (b) a file similar to the data requested. Then the sender/computer sends the principal and auxiliary digests to the receiver/computer. Upon receiving a message with these digital digests from the sender/computer, the receiver/computer searches its network cache memory **48** for data having the principal digest. If such data is found, it uses it as if it were received from sender/computer **42** and issues a positive indication signal to the sender/computer. Otherwise, receiver/computer **46** searches its network cache memory **48** for data with the auxiliary digests. If it finds data with a digital digest substantially equal to one of the auxiliary digests, it issues a partial indication signal to the sender/computer, with a reference to the digest. Otherwise, it issues a negative indication signal to the sender/computer. Upon receiving a negative indication signal, the sender/computer sends the data. Upon receiving a partial indication signal, the sender/computer transmits the difference between the digital digest of the data required to be sent and that of the data whose digital digest was found by the receiver/computer. This transaction may also begin with the receiver/computer sending a request to the sender/computer.

A modification of the above method is possible. For example, absence of the indication signal from the receiver/computer for a predefined period of time may be considered by the sender/computer as a negative indication signal, or the digital digests for some data may be stored in the permanent storage memory of the sender/computer and obtained from there instead of being calculated immediately before the transaction. Alternatively, a plurality of data may be processed in one transaction; a digital digest is calculated for each data object and a separate indication signal issued on every digital digest. Still alternatively, receiver/computer **46** may search not only in its network cache memory **48**, but also in predefined locations in its permanent storage memory. Sender/computer **42** may add to a digest it sends to the receiver/computer information about the possible location of the data with that digital digest in the receiver/computer's permanent storage memory.

Another embodiment of the present invention is schematically illustrated in FIG. **11**. Shown is a system comprising a gateway computer or gateway **60** including an operating unit, a first memory and a processor, and a caching computer **62** including an operating unit, a first memory, a permanent storage memory and a processor, connected to the gateway **60** through any fast network connection **64**, e.g., Ethernet. Gateway **60** is connected to a wide-area packet-switched network in such a way that network packets sent between at least two other computers **42** and **46** pass through the gateway **60**. The caching computer **62** uses a part of its

permanent storage memory for network cache memory **66**. Caching computer **62** has means **68** for calculating the digital digest of data in its network cache memory **66**, and means **70** for comparison between such a calculated digital digest and a digital digest received by gateway computer **60** from the wide-area network. It should be noted that gateway computer **60** may be integrally formed with the caching computer. The caching computer may have means for storing a calculated digital digest in its first memory or permanent storage memory.

By way of example, operations which may be performed in such a system will now be described with reference to FIGS. **12** and **13**. The gateway **60**, operationally interposed between a sender/computer **42** and a receiver/computer **46**, intercepts a digital digest sent from the sender/computer to the receiver/computer, saves it in its memory, and passes it unchanged to the receiver/computer **46**. Then the gateway **60** intercepts an indication signal other than a positive indication signal issued by the receiver/computer. If there was a digest for this indication signal, the caching computer **62** searches for data with the same digital digest in its network cache memory **66**. If that digest is found, then the gateway sends the data to the receiver/computer, changes the indication signal to positive, and then passes the indication signal to sender/computer **42**.

Further, the caching computer **62** may verify a digital digest for a data object stored in its network cache memory **66** by calculating the digital digest for that data and comparing it to the digest stored in the network cache memory. The calculated digital digest may be stored in the network cache memory **66** and the data object-digital digest pair may be marked as not requiring further verification.

Another further embodiment of the present invention is schematically illustrated in FIG. **14**. It consists of a network computer system comprising at least two computers: a sender/computer **42** including an operating unit, a first memory, a permanent storage memory and a processor which is connected to a network **44**. A remote receiver/computer **46** having an operating unit, a first memory, a permanent storage memory and a processor is also connected to the network. The receiver/computer uses a part of its permanent storage memory or its first memory, or both, for network cache memory **72**. The sender/computer **42** has means **74** for calculating a digital digest for data in its memory or in its permanent storage. The receiver/computer **46** has means **76** for calculating a digital digest for data stored in its network cache memory. The sender/computer **42** also has means **78** for comparison between such a calculated digital digest and a digital digest received from the network. The sender/computer further includes means **80** for calculating the difference between two data objects, and receiver/computer **46** includes means **82** for restoring a data object from another data object and the difference between said data object being restored and said another data object.

An interaction between the sender/computer and receiver/computer according to this system is illustrated in FIG. **15**. When receiver/computer **46** is required to request data from the server or sender/computer **42**, it calculates one or more digital digests for different data objects stored in its network cache memory **72** or in its permanent storage memory. Without limiting the scope of the invention, the following data objects may be recommended: (a) a previous version of the data requested; (b) a file similar to the data requested; (c) a data set similar to the data requested, which may be generated in a first memory; (d) a large data file or database including fragments of octets, similar to the data requested.

The receiver/computer then transmits a request for data, containing one or more of the above-mentioned digital digests. The sender/computer prepares a response to the request, and then calculates a digital digest on the data in the response. If the calculated digest is equal to one of the digital digests in the request, the sender/computer sends a confirmation. Otherwise, the sender/computer may continue searching for the data objects with the same digital digests in the predefined subset of its permanent storage memory. If it finds such data, it calculates the difference between this data and the data in the response, and sends only the difference. Otherwise, the sender/computer sends the response as prepared.

Variations of the above method are envisioned. For example, a number of requests for data may be sent simultaneously. The digital digests on the receiver/computer may be calculated earlier and stored in the permanent memory of the receiver/computer. The digital digests on the sender/computer may also be calculated earlier and stored in the permanent memory of the sender/computer.

It will be evident to those skilled in the art that the invention is not limited to the details of the foregoing illustrated embodiments and that the present invention may be embodied in other specific forms without departing from the spirit or essential attributes thereof. The present embodiments are therefore to be considered in all respects as illustrative and not restrictive, the scope of the invention being indicated by the appended claims rather than by the foregoing description, and all changes which come within the meaning and range of equivalency of the claims are therefore intended to be embraced therein.

What is claimed is:

1. A system for data access in a packet-switched network, comprising:

a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through said network;

said sender/computer further including means for creating digital digests on data;

said receiver/computer further including a network cache memory and means for creating digital digests on data in said network cache memory; and

said receiver/computer including means for comparison between digital digests.

2. The system as claimed in claim **1**, wherein said receiver/computer further includes means for a digital digest for data stored in said permanent storage memory.

3. The system as claimed in claim **1**, wherein said receiver/computer further includes means for storing said created digital digest in its first or permanent memory.

4. The system as claimed in claim **1**, wherein said sender/computer further includes means for the difference between two data objects and said receiver/computer further includes means for restoring a data object from another data object and the difference between said data object being restored and said another data object.

5. The system as claimed in claim **1**, wherein said sender/computer further includes means for the difference between two data objects and said receiver/computer further includes means for restoring a data object from another data object and the difference between said data object being restored and said another data object.

6. A system for data access in a packet-switched network, comprising:

a gateway including an operating unit, a memory and a processor connected to said packet-switched network

in such a way that network packets sent between at least two other computers pass through it;

a caching computer connected to said gateway through a fast local network, wherein said caching computer includes an operating unit, a first memory, a permanent storage memory and a processor;

said caching computer further including a network cache memory in its permanent storage memory, means for a digital digest and means for comparison between a digital digest on data in its network cache memory and a digital digest received from said packet-switched network through said gateway.

7. The system as claimed in claim **6**, wherein said caching computer further includes means for a digital digest for data in its network cache memory.

8. The system as claimed in claim **6**, wherein said caching computer is integrally formed with said gateway.

9. The system as claimed in claim **6**, wherein said caching computer further includes means for storing said digital digest in said permanent storage memory.

10. A system for data access in a packet-switched network, comprising:

a sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and a remote receiver/computer including an operating unit, a first memory, a permanent storage memory and a processor, said sender/computer and said receiver/computer communicating through a network;

said sender/computer further including means for creating digital digests on data, and

said receiver/computer further including a network cache memory, means for storing a digital digest received from said network in its permanent storage memory and means for comparison between digital digests.

11. A method performed by a sender/computer in a packet-switched network for increasing data access, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit data to a receiver/computer, the method comprising the steps of:

creating and transmitting a digital digest of said data from said sender/computer to said receiver/computer;

receiving a response signal from said receiver/computer at said sender/computer, said response signal containing a positive, partial or negative indication signal for said digital digest, and

if a negative indication signal is received, transmitting said data from said sender/computer to said receiver/computer.

12. The method as claimed in claim **11**, wherein said sender/computer creates said digital digest for the data before transmitting it to said receiver/computer.

13. The method as claimed in claim **12**, wherein said sender/computer transmits the data to said receiver/computer after a preset period of time has passed since transmitting said digital digest to said receiver/computer and a response signal has not been received.

14. The method as claimed in claim **12**, wherein, when a plurality of data objects is to be sent, a digital digest is sent for each of said data objects and a response signal is sent containing a separate indication signal for each of said data objects.

15. The method as claimed in claim **12**, wherein said digital digest creation comprises the step of Cyclic Redundancy Check against the contents of the data.

16. The method as claimed in claim **12**, wherein said digital digest creation comprises the step of MD5 against the contents of the data.

17. A method for increasing data access performed by a sender/computer in a packet-switched network, said sender/computer including an operating unit, a first memory, a permanent storage memory and a processor and said sender/computer being operative to transmit principal data to a receiver/computer, said method comprising the steps of:

creating and transmitting digital digests of said principal data and of one or more auxiliary data from said sender/computer to said receiver/computer;

receiving a response signal at said sender/computer from said receiver/computer, said response signal containing a positive, negative or partial indication signal, and

if a partial indication signal is received, said sender/computer transmitting a signal constituting the difference between said principal data and corresponding auxiliary data.

18. The method as claimed in claim **17**, wherein additional information about the probable location of auxiliary data in said permanent storage memory of the receiver/computer is encoded and transmitted together with the corresponding digital digest.

19. The method as claimed in claim **17**, wherein said sender/computer creates said digital digest for data before transmitting said digital digest to said receiver/computer.

20. The method as claimed in claim **17**, wherein said digital digest is obtained from the permanent storage memory of said sender/computer.

21. The method as claimed in claim **17**, wherein said digital digest creation comprises the step of Cyclic Redundancy Check against the contents of the data.

22. A method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of:

receiving a message containing a digital digest from said network;

searching for data with the same digital digest in said network cache memory,

if data having the same digital digest as the digital digest received is not uncovered, forming a negative indication signal and transmitting it back through said network; and

creating a digital digest for data received from said network cache memory.

23. The method as claimed in claim **22**, further comprising searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to the digital digest received from said network.

24. The method as claimed in claim **22**, wherein a plurality of digital digests for different data objects is received in the same message and an indication signal is generated separately for each of said data objects.

25. A method for increased data access performed by a receiver/computer in a packet-switched network, said receiver/computer including an operating unit, a first memory, a permanent storage memory, a processor and a network cache memory, said method comprising the steps of:

receiving a message containing a principal digital digest and one or more auxiliary digital digests from said network, wherein said auxiliary digital digests are correlated to data objects similar to the data object represented by said principal digest;

searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to said principal digital digest;

searching in predetermined locations in said permanent storage memory for data with a digital digest substantially identical to one of said auxiliary digital digests; and

if data having the same digital digest as one of said auxiliary digital digests received is uncovered, forming a partial indication signal and transmitting it back through said network.

26. The method as claimed in claim **25**, further comprising the step of searching the network cache memory for data with said principal digital digest.

27. The method as claimed in claim **26**, further comprising the step of searching in said network cache memory for data with one of said auxiliary digital digests.

28. A method for increased data access performed by a computer system in a packet-switched network, said computer system including a network cache memory and being operationally interposed between a sender/computer and receiver/computer so that data packets sent between said sender/computer and said receiver/computer are delivered through said computer system; said method comprising the steps of:

intercepting a message containing a digital digest transmitted from said sender/computer to said receiver/computer, and

transmitting data with a digital digest substantially identical to the digital digest received from said sender/computer to said receiver/computer in response to said message, whereby said sender/computer is relieved of the burden of transmitting said data.

29. The method of claim **28** further comprising the step of receiving said data into the network cache memory prior to intercepting the message.

30. A method for increased data access performed by a computer system in a packet-switched network, said computer system including a network cache memory and being operationally interposed between a sender/computer and a receiver/computer so that data packets sent between said sender/computer and said receiver/computer are delivered through said computer system; said method comprising the steps of:

intercepting a message containing a digital digest transmitted from said sender/computer to said receiver/computer;

intercepting a message containing an indication signal other than a positive indication signal transmitted from said receiver/computer to said sender/computer in response to said message containing a digital digest, and

transmitting data with a digital digest substantially identical to the digital digest received from said sender/computer to said receiver/computer, whereby said sender/computer is relieved of the burden of transmitting said data.

31. The method of claim **30** further comprising the step of receiving said data into the network cache memory prior to intercepting the message.

32. A method for increased data access performed by a client computer in a packet-switched network, said client computer including an operating unit, a first memory and a processor, said method comprising the steps of:

sending a request for a single first data object from said client computer to a server, said request containing multiple digital digests for different data objects similar to said first data object;

said server preparing a response to said request, searching for a second data object with a digital digest substantially identical to one of the digital digests received in said request, and producing the difference between said first data object and the uncovered second data object.

33. The method as claimed in claim **32**, further comprising the step of transmitting said difference to said client computer.

34. The method as claimed in claim **33**, further comprising the step of using said difference for restoring the data from said response in said client computer.

* * * * *

PTO/SB/44 (09-07)
Approved for use through 08/31/2013. OMB 0651-0033
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

(Also Form PTO-1050)

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

Page _____ of _____

PATENT NO. : 6757717

APPLICATION NO.: 09/398,007

ISSUE DATE : 06/29/2004

INVENTOR(S) : Goldstein Leonid

It is certified that an error appears or errors appear in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

10:47: Insert "calculating" between "for" and "a"

10:53: Insert "calculating" between "for" and "the"

11:08: Insert "calculating" between "for" and "a"

11:14: Insert "calculating" between "for" and "a"

11:19: Insert "calculated" between "said" and "digital"

11:66: Insert "calculation" between "MD5" and "against"

MAILING ADDRESS OF SENDER (Please do not use customer number below):

Proxyconn, Inc.
3211 S. Shannon Str.
Santa Ana, CA 92704

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 6,757,717 B1
APPLICATION NO. : 09/398007
DATED : June 29, 2004
INVENTOR(S) : Leonid Goldstein

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 10: line 47: Insert --calculating-- between "for" and "a"

Col. 10: line 53: Insert --calculating-- between "for" and "a"

Col. 11: line 08: Insert --calculating-- between "for" and "a"

Col. 11: line 14: Insert --calculating-- between "for" and "a"

Col. 11: line 19: Insert --calculated-- between "said" and "digital"

Col. 11: line 66: Insert --calculation-- between "MD5" and "against"

Signed and Sealed this
Fourteenth Day of August, 2012

David J. Kappos

David J. Kappos
Director of the United States Patent and Trademark Office

United States Court of Appeals
for the Federal Circuit

Microsoft Corporation v. Proxyconn, Inc., 2014-1542, -1543

CERTIFICATE OF SERVICE

I, Elissa Matias, being duly sworn according to law and being over the age of 18, upon my oath depose and say that:

Counsel Press was retained by HARNESS DICKEY & PIERCE, P.L.C., Attorneys for Cross-Appellant to print this document. I am an employee of Counsel Press.

On **September 22, 2014** counsel has authorized me to electronically file the foregoing **Brief for Cross-Appellant** with the Clerk of Court using the CM/ECF System, which will serve via e-mail notice of such filing to all counsel registered as CM/ECF users, including any of the following:

John D. Vandenberg (principal counsel) Stephen J. Joncus Carla Todenhagen Klarquist Sparkman, LLP One World Trade Center, Suite 1600 121 SW Salmon Street Portland, OR 97204 503-595-5300 john.vandenberg@klarquist.com stephen.joncus@klarquist.com carla.todenhagen@klarquist.com Counsel for Appellant	Nathan K. Kelley (principal counsel) Robert J. McManus Scott Weidenfeller U.S. Patent and Trademark Office Office of the Solicitor PO Box 1450, Mail Stop 8 Alexandria, VA 22213 571-272-9035 nathan.kelley@uspto.gov robert.mcmanus@uspto.gov scott.weidenfeller@uspto.gov Counsel for Intervenor

Paper copies will also be mailed to the above principal counsel at the time paper copies are sent to the Court.

Upon acceptance by the Court of the e-filed document, six paper copies will be filed with the Court, via Federal Express, within the time provided in the Court's rules.

September 22, 2014

/s/ Elissa Matias
Counsel Press

CERTIFICATE OF COMPLIANCE WITH TYPE-VOLUME LIMITATION, TYPEFACE REQUIREMENTS AND TYPE STYLE REQUIREMENTS

1. This brief complies with the type-volume limitation of Federal Rule of Appellate Procedure 32(a)(7)(B).

__X__ The brief contains 14,172 words, excluding the parts of the brief exempted by Federal Rule of Appellate Procedure 32(a)(7)(B)(iii),or

____ The brief uses a monospaced typeface and contains _____ lines of text, excluding the parts of the brief exempted by Federal Rule of Appellate Procedure 32(a)(7)(B)(iii).

2. This brief complies with the typeface requirements of Federal Rule of Appellate Procedure 32(a)(5) and the type style requirements of Federal Rule of Appellate Procedure 32(a)(6).

__X__ The brief has been prepared in a proportionally spaced typeface using MS Word 2013 in a 14 point Times New Roman font or

____ The brief has been prepared in a monospaced typeface using _____ _____in a ___ characters per inch_____ font.

September 22, 2014
Date

/s/ Matthew L. Cutler
Matthew L. Cutler
HARNESS DICKEY & PIERCE, P.L.C.
Attorney for Cross-Appellant